Probabilistic models for prosthetic control based on EMG decomposition

Konstantin Akhmadeev

UNIVERSITE BRETAGNE LOIRE MATHSTIC

UNIVERSITÉ DE NANTES

THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE NANTES
COMUE UNIVERSITÉ BRETAGNE LOIRE

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Génie informatique, automatique et traitement du signal, section CNU61*

Par

Konstantin AKHMADEEV

Modèles probabilistes fondés sur la décomposition d'EMG pour la commande de prothèses

Thèse présentée et soutenue à Laboratoire des Sciences du Numérique de Nantes, le 20/11/2019
Unité de recherche : Laboratoire des Sciences du Numérique de Nantes (LS2N)
Thèse N° :

Rapporteurs avant soutenance :

Christine Servière	Chargé de Recherche CNRS, GIPSA-lab, Grenoble
Frédéric Marin	Professeur, Université de Technologie de Compiègne

Composition du Jury :

Président :	Yann Péréon	Professeur, Université de Nantes
Examinateurs :	Dario Farina	Professeur, Imperial College London
	Yann Péréon	Professeur, Université de Nantes
	Yannick Aoustin	Professeur, Université de Nantes
	Éric Le Carpentier	Maître de Conférences, École Centrale de Nantes
Dir. de thèse :	Yannick Aoustin	Professeur, Université de Nantes
Co-encadrant :	Éric Le Carpentier	Maître de Conférences, École Centrale de Nantes

Invité(s) :

Antoine Nordez	Professeur, Université de Nantes

To my family and friends

Acknowledgement

Firstly, I would like to express my sincere gratitude to my supervisors Yannick Aoustin and Eric Le Carpentier. It was thanks to them that I could come to Nantes and start working here in the first place. Their support during these four years helped me a lot when I felt overwhelmed by all the questions that arise in a scientific study. I would like to thank Yannick Aoustin for his kindness, patience, and for those multiple moments when he convinced me to do things that I did not believe in at first but that turned out to be extremely helpful and interesting later. I would like to thank Eric Le Carpentier for being always so well tempered, as well as for his ability to process my fuzzy ideas and translate them into clear problem statements. I learned a lot from both of you.

My colleague and teammate, Tianyi Yu, thank you for your kindness and readiness to work. We have done very nice job together and I hope we will be able to continue that way.

I would like to thank Yann Péreon for his professional advice and the example of an infinite devotion to the work. The half day I spent observing your practice in Hôpital Nord Laënnec is one of the highlights of my thesis. Your Myostorming days are the great example of how strong professional communities are built.

I'm very grateful to Dario Farina, who was the first to introduce me to my subject, five years ago in Gottingen, Germany. Your attention and support during these years have always encouraged me to try new ideas, some of which have finally become the most important parts of my thesis.

For their professional advice and support, I would like to thank Raphaël Gross, Antoine Nordez, Philippe Jouandeau and Pierre-Antoine Gourraud. My very special gratitude is to Mathieu Courcelle for his immense help setting up the experiments.

I am very grateful to my family for their support during these four years. My mom, dad, brother, grandparents - everyone - have never questioned my choices and always cheered me. I couldn't do it without you. I also thank my dear friends Gleb Degtyaryov, Anatole Zaprudnov, Jurii Panyov and Bogdan Khomutenko for their unceasing support. You're a great part of my life, even across such distance. My best wishes to all my friends in Russia with whom we managed to stay in touch during these years, as well as to those with whom we, unfortunately, didn't.

Working in IRCCyN and then in LS2N, I have met plenty of doctorants, master and bachelor students with whom I had and, I hope, will have the greatest experiences in my life. I cannot name all of you here, guys, but if you are reading this, know that I'm talking about you and please receive my warmest greetings.

Abstract

Nowadays, robotic prostheses of lower and upper extremities draw more and more of academic and commercial interests. Such devices have the potential of significantly improving the everyday live of the amputees. This is due to the growing versatility and robustness of the robotic arms, forearms, and legs, including numerous low-cost 3d-printable designs that emerged in the last few years.

However, the wide functionality of the robotic part of a prosthesis requires an adequate interface with the user. That is, a device that replaces a missing limb cannot be controlled in the same way as a computer or a smartphone, i.e. using a keyboard or voice commands. In order for the control to be comfortable and easy, it should be perceived by the user as natural, require a small mental effort and provide fast responses to the commands with a possibility of an immediate adjustment. Thus, prosthetic control should be based on a different source of information about the command desired by the user.

The primary solution for the aforementioned problem is interfacing with the neural system of the user. The desired command, or *intent*, originates in the motor cortex and, via several intermediate levels, descends to the spinal cord, where it is transformed into excitation-inhibition patterns applied to the muscles by means of the activity spinal motor neurons. Thus, information about the intent can be extracted from the three levels: supraspinal (motor cortex), spinal and muscular. The first two, being actively studied in academia and, lately, by private companies, remain impractical since they require complex invasive procedures. Muscular level, in contrary, is much easier to interface, due to such tool as electromyography (EMG).

EMG is a recording of the electrical activity of muscles that accompanies their contraction. It permits to assess the contraction patterns of residual muscles of the stump, which can be then associated with the corresponding intent. There exist numerous EMG-based approaches to prosthetic control and new ones are being continuously proposed in academia. The majority of these approaches are based on the so-called surface EMG (sEMG), i.e. the EMG acquired by electrodes applied to the skin. In the first part of this work, we have conducted an experimental study of a classic sEMG-based gesture classification approach in order to better understand its advantages and limitations. The general relationship between the movement and sEMG has also permitted us to apply a similar technique to detection of multiple sclerosis in humans.

However, applicability of sEMG in the prosthetic control is limited by such factors as loose electrical contact between the electrode and the skin, skin and muscle

conductivity variation caused by sweat and fatigue, and variation of electrode's position relative to the muscle. These factors can be partly alleviated using *EMG decomposition*, a technique that decodes the activity of spinal motor neurons. This, being the most direct representation of the neural command, is of interest in modern prosthetic control.

EMG decomposition is widely used outside of prosthetic control, e.g. in studies of the motor system and neurology. Usually, it is applied to pre-acquired signals in an *a posteriori* setup. Although, during the last few years, a number of *real-time* decomposition methods were presented in the literature. The real-time aspect makes this technique applicable to prosthetic control. However, due to the computational complexity of decomposition task, these methods propose simultaneous decoding of only a few motor neurons (up to ten), which may be considered insufficient for establishing a precise prosthetic control.

The main interest of the present thesis is exploring the ways to extract maximum possible information from this limited decomposition, in order to infer the underlying intent. As we will show later in the text, a model-based approach that uses the known relation between the intent and the firing behavior of motor neurons can provide promising results for decomposition-based prosthetic control.

To provide fully controlled conditions for the tests of the proposed approach, we have developed a simulation model of EMG and muscle contraction process. This model yields a number of additional features that may be of interest in other studies of motor control, such as a detailed simulation of multichannel recordings by electrodes that change their position in the muscle.

Tests carried out on the base of both simulated and experimental data show that the model-based approach to the intent inference from decomposition can outperform an existing commonly used one, especially when the number of decoded motor neurons is small. These results strengthen the potential of the decomposition-based approaches being applied in prosthetic control.

Résumé

Actuellement, des prothèses robotisées de bras suscitent de plus en plus d'intérêt académique et commercial. Ces systèmes ont le potentiel d'améliorer considérablement la vie quotidienne des amputés. Cela est dû aux capacités croissantes des bras et jambes robotiques, y compris les nombreux modèles d'impression 3D bon marché, qui sont apparus lors des dernières années.

Ces prothèses mécaniquement très sophistiquées nécessitent une interface appropriée avec l'utilisateur. C'est-à-dire que la prothèse qui remplace un membre manquant ne peut pas être pilotée de la même façon qu'un ordinateur ou smartphone : par l'intermédiaire de clavier ou commande vocale. Pour l'utilisateur, la commande doit être naturelle, ne pas exiger de forte concentration mentale et fournir la possibilité d'ajustement immédiat.

La solution principale pour le problème mentionné ci-dessus est l'interface avec le système nerveux de l'utilisateur de la prothèse. La commande désirée, ou *intention*, provient de cortex moteur du cerveau, descend à travers de multiples couches intermédiaires vers la moelle épinière, où elle est transformée en excitations et inhibitions appliquées aux muscles. Ainsi, l'information sur l'intention peut être extraite à partir de trois niveaux différents : supra-vertébrale, vertébrale et musculaire. Les deux premiers, étant un des intérêts de la recherche actuelle, restent toujours difficilement réalisables en pratique. Le niveau musculaire, au contraire, est plus facile à connecter grâce à l'électomyographie (EMG).

EMG est un enregistrement de l'activité électrique des muscles qui accompagne leur contraction. Elle caractérise la contraction des muscles résiduels du moignon et peut être associée à l'intention correspondante. Il existe de nombreuses méthodes de pilotage de prothèse fondées sur les signaux EMG, ainsi que des nouvelles méthodes apparaissent régulièrement. La plupart de ces approches sont fondées sur l'EMG de surface (sEMG), l'EMG enregistré par une électrode placée sur la surface de peau. Dans la première partie de cette thèse, nous avons réalisé une étude expérimentale sur la reconnaissance des gestes classique fondée sur la sEMG, pour mieux comprendre ses avantages et inconvénients. Le lien général entre l'EMG et le mouvement nous a permis d'appliquer une méthode similaire à la détection de la sclérose en plaques chez les patients.

Par contre, le potentiel du sEMG dans le pilotage des prothèses est limité par des facteurs spécifiques comme l'instabilité de contact entre l'électrode et la peau, la variation de la conductivité de la peau entraînée par la transpiration et la fatigue de l'utilisateur, ainsi que par le déplacement des électrodes par rapport aux muscles.

Ces facteurs peuvent être partiellement réduit par la *décomposition d'EMG*, une technique qui décode les activités des moto-neurones de la moelle épinière. Ces activités, étant les représentations les plus directes de la commande neuronale, sont intéressantes pour les approches modernes au pilotage des prothèses.

La décomposition d'EMG est largement utilisée hors du domaine de pilotage des prothèses, par exemple pour l'étude du système moteur et en neurologie. Normalement, elle est appliquée aux signaux déjà acquis, pourtant, lors des dernières années, plusieurs méthodes de décomposition en temps réel ont été proposées. L'aspect temps réel rend cette technique applicable au pilotage des prothèses. Cependant, du fait de la complexité des calculs engendrés, ces méthodes ne fournissent que la décomposition d'un nombre limité de motoneurones simultanément actifs. Cette information peut être considérée insuffisante pour représenter la commande neuronale et, par conséquent, pour être utilisée dans le pilotage des prothèses.

L'intérêt principal de cette thèse est l'étude des méthodes d'extraction du maximum d'informations possible sur la commande neuronale à partir de la décomposition limitée. Nous allons montrer qu'une méthode basée sur la modélisation de la relation entre l'activité d'un motoneurone et l'intention, peut fournir des résultats prometteurs, qui soutiennent l'idée du pilotage fondé sur la décomposition.

Afin d'étudier la méthode proposée dans un environnement contrôlé, nous avons développé un modèle de simulation d'EMG et de contraction de muscle. Ce modèle possède plusieurs caractéristiques supplémentaires qui peuvent être utiles dans autres études du système moteur, telles que la simulation des signaux EMG multi-canaux, ainsi que l'EMG dans des électrodes qui changent leur position dans le muscle avec le temps.

Les tests exécutés avec les données simulées et expérimentales montrent que la méthode proposée d'inférence d'intention à partir de la décomposition peut surpasser les méthodes qui existent, en particulier quand le nombre des motoneurones décomposés est très bas. Ces résultats renforcent le potentiel des méthodes fondées sur décomposition dans les applications de pilotage des prothèses.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Application of the electromyography (EMG) to upper limb prosthetics is widely studied nowadays. This is due to the fact that control of a modern prosthetic device is required to be intuitive, robust and provide wide information flow between the user and the prosthesis. A way to ensure that is to establish a system that would autonomously infer user's *intent* and pass it to the mechanical part of the prosthesis to execute it. EMG possesses some of the important features that can help meet these requirements.

The requirements of intuitiveness and wide information flow are dictated by increasing capabilities of the mechanical part of the prosthesis, which now provides larger functionality than existing prosthetic control approaches can cover. As an example, the individual finger manipulation in robotic hands was achieved a long time ago [1, 2, 3], while the inference of corresponding intent from the user still stays a problem that is not solved.

The EMG-based intent inference may be also considered in a more general context of human-machine interfaces (HMI). There is a growing interest in versatile HMIs, which brings in larger commercial interest to the field. Private companies, such as Emotiv[1], Neuralink[2], and open-source projects, such as OpenBCI[3], engage more and more in the development of HMIs oriented to both prosthetic and non-prosthetic use. The potential of EMG in such systems is supported by the commercial success of a wearable gesture recognition device Thalmic MYO[4].

Physiological manifestations of user intent may be captured on three different levels: first is motor cortex, where the planning and onset of the motion is performed; the second level is the spinal cord, which transforms the high-level input command to low-level excitation-inhibition patterns; and the third level are the muscles, where the incoming motor command is transformed into the mechanical action.

Methods of interfacing the first level are being actively developed in the domains of neurorehabilitation and neuroscience. As an example, electroencephalography

[1] Emotiv official website

[2] Neuralink official website

[3] OpenBCI website

[4] Thalmic labs has ceased the production of MYO armbands and closed the corresponding website. We provide a link to MYO programming packages instead.

(EEG) is a technique of recording the electrical activity of the brain using a grid of non-invasive electrodes attached to the skull skin. A significant number of studies on EEG-based BCIs were made, proposing a cursor control [4], motion intent classification [5], and others [6]. The reasons why EEG-based interfaces are not being used in commercial prostheses are, first of all, a requirement for the user to be highly concentrated on the performed task, as well as a low number of recognizable patterns, and the fact that constant wearing of an array of electrodes is generally impractical. Invasive brain and spinal cord interfaces that could overcome these limitations are yet far from being widely used in humans [7]. However, a recent white paper by Neuralink [8] suggests that basic invasive brain interfaces can be accessible on the market in a few years.

At the moment, among the three mentioned interfacing approaches, muscle level provides the most promising results. This is due to the fact that the user's intent ultimately transforms into the muscle contraction process, which can be characterized by the accompanying EMG. Techniques of EMG acquisition are far less demanding than brain or spinal cord interfacing, which makes EMG a promising information source in prosthetic control.

EMG-based prosthetic control is being studied in academia and industry for already 60 years, with an extensive growth of popularity within the last three decades. However, there is a number of problems and limitations that were not yet overcome. First, the control strategy that practically reaches the amputees is still very similar to that used in the first myoelectric prostheses created in 1960s. Known disadvantages of this approach coupled with the imperfections and expensiveness of the robotic parts lead to a high rejection rate among the amputees [9, 10]. A complex development of new control strategies, more comfortable stump sockets and reliable end-effectors is still needed.

Another limitation of the EMG-based prosthetic control is the fact that in all variety of amputation cases, muscles of interest may be missing. For example, in the case of transhumeral amputation, none of the forearm muscles, which actuate the hand, are present, and therefore a source of information about the intent of hand motion is missing. One of the ways to overcome this limitation is to remap the patterns used in intent inference, to other muscles, such as those of the lower limb [11]. Another solution is targeted muscle reinnervation [12], a surgical method involving the relocation of amputated limb's nerves into a healthy muscle in order to produce intent-related EMG.

In this chapter, we will cover the physiological and technical aspects of EMG, as well as the organization of motor control. Then, we will discuss existing EMG-based strategies of prosthetic control and their classification. Further, the potential roles of intramuscular EMG and EMG decomposition will be analyzed. That will bring us to the problematic of this thesis, which is a decomposition-based inference of user intent.

1.1 Basic notions of EMG, motor control and EMG decomposition

EMG is a recording of the electrical activity of muscle fibers that naturally accompanies their contraction during the execution of motor tasks. The contraction of muscle fibers is regulated by *alpha motor neurons* (MN) of the spinal cord, which, in their turn, are defined by supra-spinal commands originating in the motor cortex, and are regulated by numerous inter-spinal mechanisms of excitation, inhibition, and feedback. EMG is, therefore, a function of the total neural input to the muscle. In prosthetic applications, this makes EMG a great source of information about the user's intent. In this section we will shortly cover the main point about the physiology and modelling of EMG.

1.1.1 EMG signal generation

A motor unit (MU) is an elementary entity of the motor system. It consists of a MN located in the spinal cord and of a set of muscle fibers innervated by its axon [13]. The last originates in the soma of MN and reaches the specific muscle or muscle group as a part of an efferent nerve. The axon then separates from the nerve in the vicinity of the muscle and forms a so-called terminal arborization to innervate a certain number of muscle fibers, usually referred to as *muscle unit* (MsU).

As a nerve cell, MN responds to excitatory input by a series of its cell membrane depolarizations. The last propagate through the axon in form of *action potentials* (MNAP) up to the junction with the muscle fibers. Due to the specific interfacing mechanism between an axon and a muscle fiber, called neuromuscular junction, MNAP excites an action potential in the fiber (SFAP). SFAP travels along the fiber, causing its contraction until it reaches the tendon and fades.

All fibers belonging to a single MU receive the MNAP almost synchronously with specific delays determined by the differences in their axon branch lengths. This creates a unique spatial configuration of SFAPs, which, propagating along fibers with different velocities, further de-synchronize with each other.

Each SFAP causes the variation of the electric potential field around the muscle fiber, which can be detected by a metal electrode placed in its vicinity. Group of SFAPs evoked by the same MNAP produce a potential variation that can have unique temporal form when acquired by the electrode. The last also depends on the mutual positioning of the electrode and muscle fibers, their shapes and sizes. These shapes, as they are recorded by the electrode and the following acquisition system, are called motor unit action potentials, MUAPs.

While receiving excitation, MN generates MNAPs in a rhythmic fashion, with intervals between depolarizations of about 25 ms to 200 ms. A group of multiple successive action potentials is usually referred to as *spike train*. Since fibers of MUs are highly intermingled, an electrode usually records MUAPs from multiple MUs at the same time. Each MUAP is caused by a MNAP of corresponding MN. Thus, EMG is a mixture of repetitively appearing MUAPs of several different MUs located

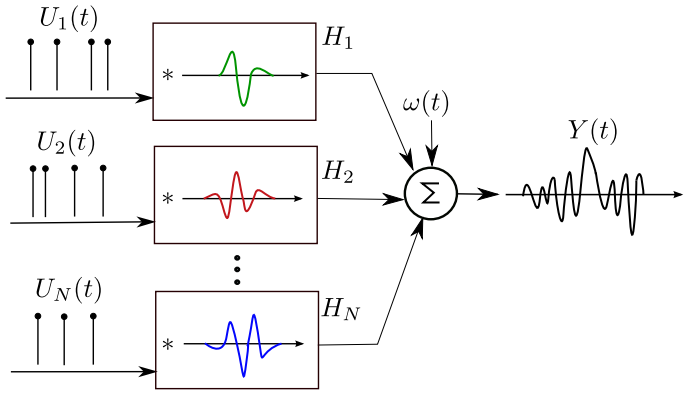


Figure 1.1: Illustration of mathematical model of EMG signal 1.1.

in the recording area of the electrode.

Despite being slightly out of physiological context, commonly used linear model of EMG helps to the understanding of this process [14, 15]:

$$Y(t) = \sum_{i}^{N} H_i * U_i(t) + w(t) \tag{1.1}$$

where N is the total number of MUs, $U_i(t)$ is a sequence of Dirac functions representing the spike train of i-th MU, H_i is MUAP of i-th MU, symbol $*$ denotes convolution, and $w(t)$ accounts for instrumentation noise of the acquisition system, as well as for interferences recorded by the electrode. This model is illustrated in Figure 1.1.

1.1.2 EMG electrodes

EMG electrodes are metallic bodies (usually silver or stainless steel) in an isolation housing, destined to be in physical contact with the skin or the muscle fibers. The size of the electrode and configuration of the front-end signal processing circuits define its recording volume. SFAPs of fibers located within the recording volume contribute to the total acquired EMG.

EMG electrodes are first classified as *surface* (sEMG) and *intramuscular* (iEMG). The former are applied to the skin, while the latter are inserted subcutaneously into the muscle belly. sEMG electrodes are usually larger than iEMG ones, covering a considerable part of a muscle and thus having larger acquisition volume. This makes sEMG useful when the global activity of the muscle is of interest, since it integrates signals from multiple MUs at the same time. However, this also comes with the inability of sEMG to record solely from a target muscle, capturing the activity of neighboring ones. This effect is usually referred to as *cross-talk*. The fact that the electrode is positioned on the skin also means that MUAPs should travel across skin tissues before being recorded, which introduces a considerable low-pass filtering effect to the EMG.

iEMG electrodes, due to their subcutaneous nature, are small and therefore record from a much smaller volume, permitting to target specific regions in the muscle. Moreover, such electrodes can reach deeply-lying muscles that couldn't be accessed by sEMG, such as pronator and supinator of the hand. Muscle tissue has better conductive properties than skin and fat, which makes iEMG preserve the high frequency components of the MUAPs. However, invasiveness of iEMG brings risks of infection, pain, and discomfort, which until recently made iEMG a very specific choice, dedicated mostly to neurological studies, rather then to prosthetics.

The aforementioned features of sEMG and iEMG are illustrated in Figure 1.2. Further classification of the electrodes is mostly based on their sizes and multi-channel arrangements. That is, the smallest intramuscular electrodes record from a single muscle fiber (SFEMG) [16]. Fine-wire electrodes record from a slightly larger volume, capturing several MUs at considerable contraction levels. Monopolar and concentric bipolar needles record from larger volumes of 0.02 - 0.1 mm^3 detecting 10-15 MUs even at moderate contractions. A subsequent increase in recording volumes is achievable using multichannel intramuscular recordings [17]. Long-term recordings, including prosthetic control applications, can be achieved using chronically implantable acquisition systems [18, 19, 20, 21, 22] (more on this type in Section 1.2.5).

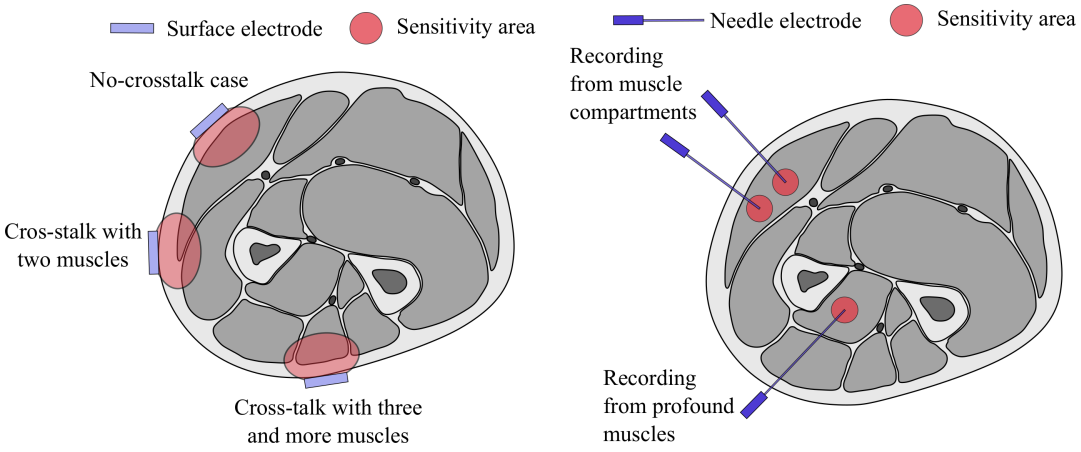
sEMG electrodes can be classified accordingly. Myoelectric pattern recognition approaches (see Section 1.2.2) widely use several (usually up to eight) sEMG electrodes arranged around the wrist. Matrices of small sEMG sensors, arranged with high density (HDEMG) are more and more used in academia for prosthetic control, providing very detailed information on the spatial distribution of EMG activity across the muscle.

The entire variety of existing EMG electrodes permits to choose the characteristics of the signal that are best adapted to the specific problem. Some solutions, like intramuscular recordings, are less practical than others when considered in the context of prosthetic control. However, with the constant advance of technology, new long-term intramuscular acquisition systems emerge, making iEMG a more and more realistic choice for myoelectric control. At the same time, the full potential of conventional sEMG is not yet realized in the existing prostheses.

1.1.3 Motor control

Motor command originates in the motor cortex of the brain. Then, across several intermediate levels such as midbrain, pons, and medulla, it "descends" to the spinal cord. Along this path, it is transformed into excitation/inhibition patterns, which are applied to the MNs of the spinal cord in order to contract or relax specific muscles, ultimately leading to the execution of the intended movement.

Force of muscle contraction is modulated by two processes: recruitment/de-recruitment of MNs and variation of their firing rates. An MN is recruited (i.e., starts firing) when the amount of its excitatory input reaches a so-called *recruitment threshold*, specific to each MN. After recruitment, the following increase of the excitatory input leads to the increase of firing rate. That is, the motor system



(a) Spatial selectivity of surface electrodes

(b) Spatial selectivity of intramuscular electrodes

Figure 1.2: Schematic illustration of spatial selectivity of intramuscular and surface EMG electrodes. Sensitivity areas are not in scale and may vary for different electrodes. Muscle cross-section reproduced from Gray's Anatomy [23].

regulates the muscle contraction force using excitatory input to its MNs.

The entirety of MNs that innervate a single muscle is called a *motor neuron pool* (MNP). MNs belonging to the same pool differ from each other by their size. That is, the physical size of MN's body defines its behavior in response to the neural input [24], as well as the relative portion of the muscle that it innervates. That is, larger MNs tend to have larger recruitment thresholds, i.e., during an increasing contraction they are recruited *later* than the smaller ones. This phenomenon is usually referred to as Henneman's principle [25].

The excitation-rate curve of MN is also defined by its size. Commonly used experimentally supported assumption is that at each level of contraction, firing rates of smaller MNs are greater than those of the larger ones [26, 27]. This phenomenon is usually referred to as *onion-skin* principle, since, if traced on the same axis, firing rates of smaller MNs tend to always stay above those of larger ones.

It is common to represent an MNP as a functional unit, in which all the MNs receive a common excitatory input, dedicated to elicit the contraction of the corresponding muscle [28]. Experimental studies show that variations of firing rates of MNs are highly correlated during changes in the contraction force [26, 28]. That is, there is some degree of synchronicity with which MNs change their firing rates. In order to explain this phenomenon, C.J. De Luca and Z. Erim have proposed the notion of *common drive* [28], i.e., the existence of a single source of excitation for all the MNs of a pool.

At first, the common drive was associated only the with supra-spinal command, i.e, was supposed to be determined solely by the motor cortex. Later, it was shown that the firing rate correlation is influenced by proprioceptive feedbacks (muscle spindles and Golgi tendon organs) [29, 30] as well as by antagonistic and agonistic muscles contractions [31, 32]. Considering more recent models [33] of motor control

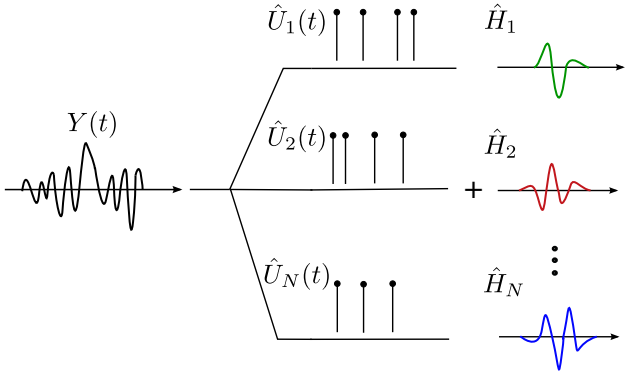


Figure 1.3: Illustration of EMG decomposition process as inverse of model 1.1 and Figure 1.1. $\hat{U}_i(t)$ and $\hat{H}_i$ are the estimates of spike train and MUAP of i-th MU.

organization on the spinal level, it may be also influenced by inhibitory feedback introduced by Renshaw cells.

Common drive is considered to be a physiological tool that facilitates the motor control for CNS. It is assumed to provide centralized control over an entire MNP instead of independent control over each MN individually [34]. In the modelling of the motor control, the common drive is a useful notion that permits to define the input of the MN pool as a time-varying scalar function [35]. In different studies, it may be referred to as *common synaptic input* or *net excitation*.

As we can see, EMG is a function of spike trains produced by MNs and therefore is a function of the excitatory input to the MNP. That is, EMG characterizes the excitatory input that causes muscle contraction. In the case of amputation, this information, based on the knowledge of the biomechanical function of the muscle, can be used to infer the user's intent: which movement he or she wants to be performed and with which stiffness or speed, which constitutes the problem of prosthetic control. The following section will provide the necessary terminology and classification of existing prosthetic control approaches.

1.1.4 EMG decomposition

As we have described earlier in this section, muscle contraction is the result of the spiking activity of MNs that innervate this muscle. Studies in such fields as neurology, motor control, biomechanics and, lately, myoelectric control, require the assessment of this activity. Direct recordings *in vivo* from MNs ' somas or axons are possible, but the complexity of such procedures limits their application in practice. Instead, spiking activity can be *inferred* from the EMG signal.

Expression (1.1) serves as a good illustration of this process. That is, we observe an amount of EMG signal $Y(t)$, $t \in [0, T]$ and we want to estimate the following: the number of MUs that contribute to the signal N, their MUAPs H_i and spike trains $U_i(t)$. This process is termed the *decomposition* of EMG. Figure 1.3 illustrates is with respect to the model in Figure 1.1.

At first, it may seem that there's an infinite amount of solutions to this problem,

i.e., combinations of H_i and U_i that satisfy the equation (1.1). However, there are some physiology-based constraints that one can apply to the possible solutions. First of all, spike trains $U_i(t)$ exhibit sparsity and regularity. That is, the intervals between spikes are much longer than the duration of a MUAP, and the spikes, during steady or slowly varying contractions, appear in a rhythmic fashion. Second, the forms of MUAPs do not change in time or change much slower compared to the frequency of their appearance, so that they can be efficiently tracked.

Decomposition can be accomplished either manually as well as semi- or fully automatically. In the first two cases, such tools as EMGLAB [36] can be used [1]. As for the automatic decomposition, there exist numerous techniques, the most influential of which will be covered in this section.

Historically, the first approaches addressed the decomposition of iEMG, that is much less cluttered compared to sEMG. In [37] and [38], template matching and clustering algorithms were applied to detected MUAPs in order to sort them into distinct groups with no attempt of superposition decomposition. In [39], prior knowledge on spike regularity is used and decomposition of at most two superimposed MUAPs is proposed. Automatic detection threshold adjustment, sub-sample level MUAP form matching, and peeling-off of the greater MUAPs before subsequent re-decomposition were described in the ADEMG technique [40]. Adaptive MUAP clustering technique was proposed in [41].

Commonly acclaimed EMGLAB package [36] includes a modified version of ADEMG, as well as more advanced algorithm for multichannel iEMG decomposition MTLEMG [42]. The last one extracts features from the MUAP shapes and clusters them in order to obtain initial estimates of MUAPs and MUs. Then, clusters that have similar forms and matching firing timings are merged. Estimated MUAPs are then subtracted from the signal and another detection and clustering phase is performed. Finally, all MUAP estimates are used to resolve superpositions.

Other techniques are also worth mentioning, such as density-based clustering [43], autoencoder-based MUAP feature extraction [44], genetic optimization for superposition resolution [45], Viterbi algorithm for spike train estimation [46].

It is also worth mentioning that the problem of iEMG decomposition is very similar to the problem of spike sorting in recordings of neuronal activity. Mostly, spike sorting techniques are based on detection and clustering phases, are variate in terms of the used features and clustering techniques. Examples of spike sorting techniques may be found in [47, 48, 49].

sEMG decomposition became possible due to the integration of information from multichannel recordings using blind source separation [50, 51, 52], including the tracking of changing MUAPs [53], as well as other techniques [54, 55]. Due to its practicality, sEMG-based decomposition becomes is used more and more commonly in the studies of biomechanics and motor control.

Due to the large variety of existing automatic decomposition methods, it may be difficult to choose one for use in a particular study. However, this choice is usually led

[1] If reader has no experience of manual EMG decomposition, a very clear interactive example is given in http://emglab.net/emglab/Software/Software.php.

by the accessibility of the decomposition technique, i.e., whether it is implemented, optimized and distributed in form of an open-source program, or as an additional package for commercial hardware or software. An example of the former is EMGLAB [36], MATLAB package for visualization and computer-aided manual decomposition of iEMG signals. Examples of the latter are OTBioeletronica's Decomponi (CKC-based decomposition of high-density EMG signals)[1], Demuse Matlab package[2] and Delsys systems[3].

1.2 Existing myoelectric control strategies

1.2.1 Myoelectric control classification and terminology

In order to clearly describe a prosthetic control strategy, a classification of existing strategies is required. In this work, we use the system proposed in [56], which we will partly provide here.

First, this classification system introduces the notion of *motor function*, which is a distinct type of prosthesis' movement, disregarding the speed or direction, e.g. hand gripping/opening, wrist flexion/extension, etc. Motor functions can involve one or several degrees of freedom (DOFs) of the mechanical part, e.g. single phalanx flexion/extension, wrist rotations, etc.

This system classifies prosthetic control strategies basing on three aspects:

- *Input sources*, the set of input information sources which can be surface EMG (sEMG), intramuscular EMG (iEMG), body or stump trajectory (e.g., from an inertial measurement unit), user input via a smartphone application and others;

- *Intent interpretation*, or how many motor functions the approach permits to control in total, and whether they can be controlled simultaneously (simultaneous control) or just one at a time (sequential control) (see Figure 1.4);

- *Activation profile*, or whether the approach permits to regulate some mechanical output, such as joint angle or angular velocity, in a continuous (proportional control) or binary way (on-off control)

Most of the existing myoelectric control strategies can be efficiently described by this classification [56]. Up to this date, the simultaneous proportional control strategy with multiple motor functions stays the main challenge of academic studies in the domain of myoelectric control. In the next sections, we will describe principal control strategies presented in the literature.

[1] https://www.otbioelettronica.it/en/products/software

[2] https://demuse.feri.um.si

[3] https://www.delsys.com/timeline/first-non-invasive-system-for-decomposition-of-surface-emg-signals/

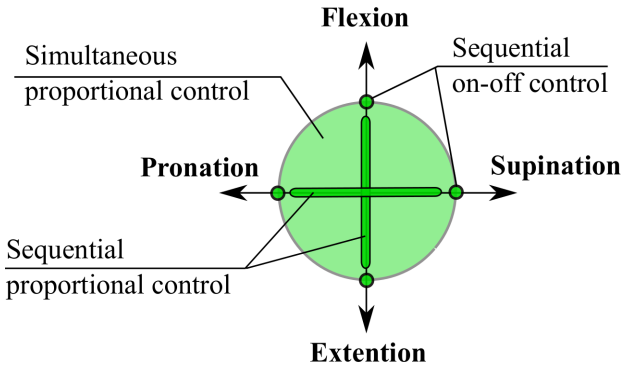


Figure 1.4: Illustration of prosthetic control taxonomy on the example of wrist extension/flexion and pronation/supination. Green zones represent attainable control commands.

1.2.2 Dual-channel amplitude-based myoelectric control strategy

The first publication on myoelectric prosthesis dates 1948 and is made by physics student at Munich University, Reinhold Reiter [57]. This potentially revolutionary concept didn't achieve clinical nor commercial use [58] until the year 1960 when Kobrinsky [59, 60] has proposed a more practical design, which became possible due to advances in transistor technology. This device was capable of performing two functions: closing and opening the hand, both triggered by activity in one of two EMG electrodes placed above a pair of antagonist muscles groups of the forearm: flexors and extensors of the wrist and fingers. EMG was amplified, low-pass filtered and rectified to produce an estimate of the signal envelope, which was then compared with a certain pre-defined threshold in order to trigger either closure or opening of the gripper. The controller was also capable of detecting simultaneous contraction in both muscles and interpreting it as a command to stop the motion.

Even nowadays, many commercial prosthetic systems are still based on this strategy [61, 62]. Improvements mostly consider the mechanical part of the prosthesis: its weight, robustness and wider range of possible movements. The control strategy, on the other hand, undergone only a few modifications. Current strategies provide the user with a choice among several grasps and gestures, although the switching between them is still triggered by the simultaneous contraction in both muscle groups. Having chosen a gesture, the user may adjust it by the contraction of one of the two muscle groups, e.g. flexion/extension for closing/opening the hand (see Figure 1.5). According to the classification provided above, this is a proportional control strategy with a more restricted sequential approach. Such a technique is robust and easy for the user to learn. However, it requires to remember which motor function is currently chosen, as well as the number of co-contractions needed to activate the next one.

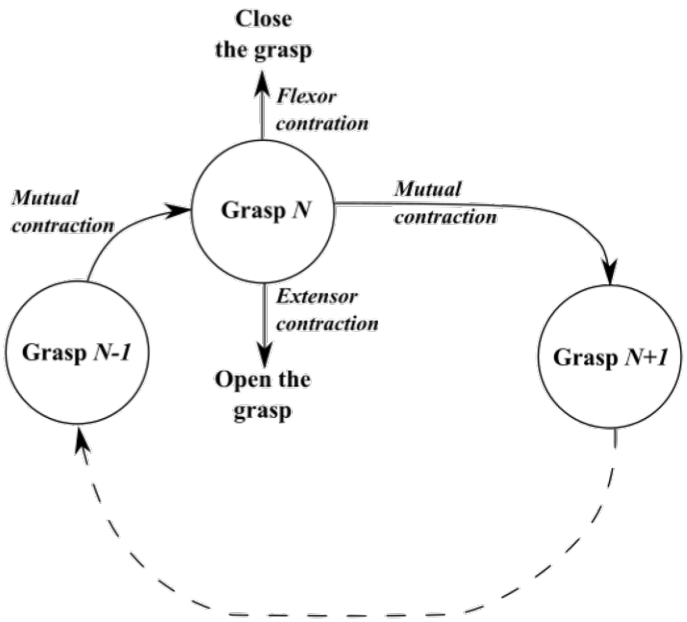


Figure 1.5: Graph representing the classic dual-channel myoelectric control strategy. The user switches to the next motor function (grip) in the graph by performing a co-contraction of antagonistic muscle groups. While a function is chosen, the user can adjust it to the current task by contraction one of the muscle groups.

1.2.3 sEMG pattern recognition myoelectric control strategy

The first attempts to avoid using the cyclic graph control were based on an increased number of sEMG channels in combination with machine learning techniques. The family of such strategies can be referred to as *myoelectric pattern recognition*. Then involve placement of several electrodes around the forearm (usually around 8) either symmetrically, or above specific muscles (targeting) [63], signal windowing, feature extraction, and subsequent supervised classification.

The variety of all possible hand movements is discretized to form an acceptable number of classes ([56, 64]). These classes are designed to cover the most important motor functions and, at the same time, to be distinguishable by a classification algorithm. The most frequently used classes are cylindrical grasp, hand rotations, wrist flexions and extensions, radial and ulnar deviations ([65]).

In order to apply classification techniques to EMG, one should first reduce its dimensionality. This is usually achieved by means of feature extraction and feature selection. Features are scalars or vectors that represent temporal or spectral parameters of the signal, taken in a short window (usually not longer than 250 ms). The number of feature types presented in the literature is large ([64]), and it is shown that some of them are more effective than others when applied to myoelectric pattern recognition [66].

Feature extraction permits to encode various aspects of the signal, such as energy,

amplitude, spectral band, etc. It is hard and in most cases not possible to bond these parameters into one physiology-based model. Thus, multivariate statistical analysis and machine learning were considered to be suitable techniques for finding such models automatically. Numerous approaches were investigated during last two decades: Linear Discriminant Analysis [67, 68], Neural Networks [69] and Support Vector Machines [70].

This strategy is essentially a sequential on-off control. Its advantage over the previous one is that the grip graph, compared to Figure 1.5 is fully connected. That is, each grip or gesture is accessible from any other grip. However, the main challenge for this approach is the fact that surface electrodes, that provide the input for pattern recognition, shift during daily activities, which causes a drop of the recognition accuracy since patterns are learned in the original position of the electrodes [71, 72].

It took almost three decades for this strategy to be recognized and implemented by companies producing myoelectric prostheses. At this moment, it is being implemented in commercial prostheses [73], [74]. Rate of acceptance of such devices by the amputees should be analyzed in the future. This is important since even the classic two-electrode strategy sometimes fails to satisfy the user when exposed to such factors as variable posture, sweat, and fatigue [75].

1.2.4 Other sEMG-based myoelectric control strategies

A strategy for simultaneous and proportional control of at least two motor functions was presented in [76]. This approach supposes that each motor function is regulated by a single excitatory input to the synergistic muscles that actuate it. Then, the authors propose a model that links these inputs to the sEMG signal in electrodes placed in the vicinity of the muscles. The model parameters are estimated from training data using non-negative matrix factorization. Training data consists of the motor function values and simultaneously acquired sEMG.

This method, since it is based on sEMG, may suffer from the electrode shifts and fatigue (as for the variation of contraction force, we note that it is included in the model). However, authors have provided a separate study [77] on that cause and, using high-density electrode matrices, have shown that both longitudinal and transversal shifts do not significantly affect the accuracy of this control strategy. This method was tested on a commercial prosthesis and have provided considerable acceleration on classic motor tests[1] .

Another variant of pattern recognition and non-negative matrix factorization arises from the use of high-density electrode matrices. As an example, for pattern recognition, such features as spatial correlation, specific to these kinds of recordings, are extracted and classified [78]. Model for non-negative matrix factorization can be adapted to high-density electrodes [77].

Further improvements of existing strategies can be achieved using physiology-based modelling of how the user's intent transforms into sEMG in each of the avail-

[1] A video example available under this link.

able channels. In [79], a neuromusculoskeletal model defined to, first, transform sEMG features to muscle contraction forces and, second, transform muscle contraction forces to a set of feasible prosthesis movements. In order to collect calibration data for the model, the user is asked to mimic prosthesis movements with his or her phantom limb. As authors suggest, neuromusculoskeletal modelling permits to constraint the derived intent to a set of feasible movements and provide more natural control to the user.

To conclude, at the moment sEMG stays the only choice for interfacing the user's motor system in commercial prostheses. Electrode shifts, being one of the most challenging issues, can in some cases be countered by choosing specific electrodes and control strategy [77]. The lack of spatial selectivity in sEMG makes it impossible to record solely from target muscles (so-called *cross-talk*), which complicates the building of the control model. Incorporation of the cross-talk into the model, as in non-negative matrix factorization approaches [76], can slacken this problem. Thus, the full potential of sEMG-based strategies presented in the literature is yet to be realized in market prostheses.

However, cross-talk and electrode shift problems can also be alleviated by using a different acquisition approach, which is, for example, *intramuscular* EMG. In the following section, let us consider existing applications of iEMG in prosthetic control.

1.2.5 iEMG-based myoelectric control strategies

Intramuscular EMG (iEMG) is an invasive method that involves the placement of the electrodes either subcutaneously or in the interior of the muscle tissue. In the prosthetic control perspective, iEMG has a number of advantages compared to much more conventional sEMG, such as high spatial selectivity, access to deeper muscles and lesser exposure to ambient noise and interference.

High spatial selectivity permits to record from a target muscle, avoiding interference from the surrounding ones. As we have mentioned earlier, the last is a common problem of sEMG-based control strategies. Moreover, intramuscular electrodes permit to acquire iEMG from deeper muscles, such as pronator or supinator of the hand.

Despite the aforementioned advantages, iEMG is considered impractical for prosthetic control due to its invasive nature that brings risks of infection, discomfort, and pain to the user. However, there exists a number of studies [18, 19, 20, 21, 22] that propose to surgically implant wireless acquisition system into the muscle. These implantable myoelectric sensors (IMES) usually have cylindrical form, with a length of 10 mm and diameter of 3 mm [18]. They consist of a single-channel analog instrumentation amplifier, pass-band filter, analog-digital converted, and a magnetic coil for the wireless battery recharging and iEMG data transmission.

In such systems, as it is shown in [21], the total sampling frequency shared by all channels may reach 14 kHz. A high sampling frequency is of interest when complex iEMG processing approaches are used, such as decomposition. Alternatively, as in a more recent study [22], the signal can be rectified and low-pass filtered on the IMES and then transmitted with a much lower sampling frequency (e.g., 67 Hz in

[22]). This is an optimal solution for strategies where rectified sEMG is used as an estimate of muscle contraction force.

Rectified and low-pass filtered EMG was already used as an estimate of the proportional intent in studies on iEMG applied to prosthetic control. For example, [80] associate each motor function with one single iEMG channel and calculates the proportional input for the prosthesis as a low-pass filtered rectified signal. In [81], a slightly different approach is proposed, in which the proportional input is estimated using the difference between the rectified signals from two antagonistic muscles.

However, the small acquisition area of intramuscular electrodes can also be a disadvantage in this case. This is due to the fact that observed muscle fibers may not represent the muscle as a whole [82]. The same authors [83] propose to use an artificial neural network to establish a non-linear regression model that relates iEMG features to the proportional input.

To conclude, in prosthetic control, such advantages of iEMG as spatial selectivity and access to deep muscles, come together with the need for complex technological solutions, such as chronic implants with wireless data transmission. However, the last ones are becoming more and more available since the technology is being explored and improved by several independent research and engineer groups.

1.2.6 Existing decomposition-based control strategies

A real-time decomposition of hdEMG and subsequent gesture classification is presented in [84]. The decomposition is achieved using an iterative version of blind source separation technique [85], reaching the decomposition of approximately up to 15 MUs. Resulting individual firing rates are then low-pass filtered and passed as an input to a support vector machine classifier, recognizing four different gestures.

The approach proposed in [86] decomposes a high-density EMG (hdEMG) of a TMR[1] subject using the convolution kernel compensation algorithm [50]. It groups decomposed MUs into four categories depending on their approximate location in the observable part of the muscle, corresponding to four quadrants of the HD electrode. Their firing frequencies, along with their categories are then used as features for a support vector machine classifier. Alternatively, in this study they use a neuromusculoskeletal model, similarly to the one we have mentioned in Section 1.2.4, that is able to transform decomposed spike trains into joint torques of a calibrated musculoskeletal model, that then are projected to plausible movements of the prosthesis.

In [87], it is proposed to estimate the common input to a motor neuron pool by combining its individual estimates obtained from the decomposed spike train of each

[1]Targeted Muscle Reinnervation (TMR) [12], a surgical technique consisting of, during transhumeral (shoulder) amputation, recovering the nerves of the amputated limb and placing them in an intact muscle, such as pectoralis major (chest). During the first months after the operation, the relocated nerve innervates the host muscle. Intended movements of the phantom limb, in that case, result in the contraction of host muscle with coincident EMG generation, which can be captured and analyzed for prosthetic control. TMR is an example of the more general concept that perceives the muscle as a biological amplifier of the spinal command.

single motor neuron. The authors use the common assumption that motor neurons of one pool receive a common input that regulates their joint production of muscle contraction force and, therefore, the mechanical parameter of the movement (in their case - the angle of wrist rotation). They directly replace the common input by angle measurements and model the angle-spike relationship as linear. Since angle estimates obtained from different motor neurons will not be identical due to the presence of individual inputs to the MNs, and decomposition errors, the authors use median value across the pool as a global estimate of the angle.

Despite not being directly related to prosthetic control, [88] studies the relationship between smoothed composite spike train (CST[1]), as a measure of neural drive to the muscle, and contraction force of a muscle. They suggest that CST is better correlated with the force, especially at higher frequencies, than convenient the integrated sEMG.

While previously mentioned approaches work with hdEMG decomposition providing spike trains of 10 up to 20 MNs, [89] aims to infer the contraction force from a smaller number of MNs (1 to 10). Such numbers of decomposed trains are usually observed in the decomposition of fine-wire or needle iEMG at low contraction forces. The authors propose to, first, estimate force-rate relationships for all decomposed MNs, then to use the inverse of these relationships to estimate force from each currently observed inter-spike interval.

This approach can be applied either to the estimation of a single muscle contraction level or to the proportional control of an entire motor function. Former will be of use in such control strategies as [79], where the contraction level of each muscle feeds a musculoskeletal model that then produces motor commands for the prosthesis. The latter application will be of use when EMG of only one muscle of a synergistic muscle group is being observed, so that the neural drive to the muscle group is estimated from this single muscle.

1.3 Problematic of decomposition-based myoelectric control

Studies of decomposition-based control strategies appear much rarer compared to those on pattern recognition or other sEMG-based approaches. This is due to the high complexity of the decomposition task itself, both in terms of implementation difficulty and of computation power that this procedure demands.

Prosthetic applications require that all the processing applied to the signal should be performed online with a maximum delay of about 250 ms [90]. Up to this moment, there's only three existing real-time full decomposition approaches [91, 84, 92] that can provide decomposition with such a delay. The first is a recursive version of the convolution kernel compensation algorithm [50]. It is able to decompose an experimental signal acquired by a 6×5 matrix of sEMG electrodes in real-time.

[1]CST (cumulative or composite spike train) is the spike trains of all decomposed MNs summed up into one train as if it was produced by one single neuron. It usually serves as an estimate of neural drive or excitation applied to the motor neuron pool.

More specifically, their signal was generated during a 10% MVC isometric contraction of the tibialis anterior muscle. The algorithm was capable of decomposing 4±2 MUs, correctly identifying at least 90% of their discharges. The second is a recursive version of blind source separation method [85], achieving decomposition of approximately up to 14 MUs at contraction levels of up to 30%MVC on a standard PC, using a matrix of 64 electrodes. Experiments were conducted on tibialis anterior and flexor digitorum superficialis muscles.

Approach presented in [93] is based on [94] and is being constantly developed by our team. It is based on state-space modelling of the MNs spiking activity, sampling of possible spiking scenarios and Bayesian filtering of the state vector. The advantages of this approach compared to the two previous ones are the use of a single-channel signal, the capability of tracking the slowly-varying changes of MUAPs' shapes and adaptation to the additive noise level. Also, single channel intramuscular EMG is potentially better adapted to the variety of muscle geometries, while approaches based on hdEMG tend to perform worse on muscles whose fibers are not parallel to each other and to the skin.

Originally not adapted to changes in the number of active MUs, our approach was recently modified to decompose iEMG of dynamic contractions [93]. Next step of its development [92] is a GPU[1]-based implementation permitting to perform real-time decomposition of up to 10 concurrently active MUs during dynamic contraction with global performance scores reaching 95%.

All of the presented methods are limited to the decomposition of a relatively small number of MUs (up to 6 in [91], 10 in [92] and 14 in [84]), which questions their applicability in prosthetic control. That is, these numbers are small compared to the total numbers of MUs even in small muscles, such as first dorsal interosseous (120 MUs [95]). Therefore, it is a question, whether such small number of MUs can be a representative subset of the entire MU pool, compared to, e.g. the rectified sEMG that integrates the activity of tens and hundreds of MUs.

In the case of [92], the number of decomposed MU is limited both by the available computational power and by the acquisition volume of the intramuscular EMG electrode. An increase in the number of decomposed MUs by running several independent instances of the algorithm on several spaced out electrodes does not seem viable, at least for prosthetic control applications. This is due to the fact that prosthesis-embedded GPU-based decomposition[2] of one single channel is a challenging task by itself and is not yet achieved. This leads us to the problematic of this research, which is: *how to estimate the level of muscle contraction or of the neural drive to the muscle, using a very limited number of decomposed MU*? More detailed reasoning about this problem will be given in the following section.

[1]GPU - Graphics processing unit, or graphic card. Originally destined to PC screen image rendering, they are now widely used for parallel computations.

[2] Possible approaches to this task may involve recently developed Nvidia Jetson Nano, a developer kit for embeddable GPU applications.

1.4 Problematic of this research

1.4.1 Potential advantages of decomposition-based myoelectric control

EMG is a function of not only neural command to the muscle. That is, it is influenced by the form and position of the electrode, electrical properties of the conducting medium, additive noise and condition of the muscle fibers, such as fatigue. All these factors can vary in time and their influence on the EMG signal is hard to predict or compensate for.

The decomposition of EMG breaks the signal down into its constituent spike trains and MUAPs, naturally separating electrode-dependent aspects of the signal (MUAPs) from the command-dependent one (spike trains). Thus, spike trains are free from disturbances that are not related to the neural command, such as electrode shifts, skin properties, and additive noise. This is with the exception of fatigue, which may affect both the MUAPs and spike trains.

Previously presented real-time decomposition algorithm [93] is capable of tracking slow variations of MUAPs that can be caused either by electrode shifts or by gradual changes in the conducting medium properties. Moreover, it is capable of adapting to the changes in the number of active MUs, thus providing a decomposition of dynamic contractions. These features are vital for establishing a stable decomposition of iEMG in prosthetic devices.

We believe that this algorithm, having strong mathematical formulation, can be further adapted and optimized in order to fit very restrictive requirements of embedded computing. In the future, with a stable on-board real-time decomposition, prosthetic control systems will be able to provide much finer simultaneous and proportional control over multiple motor functions.

1.4.2 Challenges of decomposition-based myoelectric control

Along with the aforementioned advantages, decomposition-based myoelectric control comes with challenges that need to be addressed in order for it to find an application in prosthetics. These challenges come from the limitation of the existing real-time decomposition algorithm:

- Decomposition requires much larger computational power than current prosthetic devices provide;
- Available computation power in embedded applications limits the number of decomposable MUs, thus making only low-range contractions ;
- Decomposition is prone to errors;

Also, a number of challenges come from the intramuscular nature of the signal:

- Impracticality of iEMG in daily use;

- Small portion of the muscle observed by an intramuscular electrode may not represent the muscle as a whole;

In previous sections, we have already mentioned ways to overcome these challenges, let us list them here. The impracticality of iEMG in daily use may be tackled by the usage of chronic intramuscular electrodes, such as ones described in [21]. Necessary on-board computational power can be achieved by use of GPUs dedicated for embedded applications, such as that of NVidia Jetson Nano. Errors of decomposition should be accounted for by an appropriate modelling that includes the most frequent errors, such as missed spikes, masking of small MUs and others.

The small size of the observed muscle volume and the limited number of decomposed MUs constitute, in fact, one single issue: whether decomposition can provide sufficient information on the intent. In this work, we aim to propose a physiology-based intent estimation model that extracts the maximum information from the available decomposition.

1.4.3 Physiological basis for approaches that use small number of decomposed MUs

There are several physiological phenomena that support the possibility to establish a proportional control based on the decomposition of a limited number of MUs, detected in a small region of the muscle observable by an intramuscular electrode.

First, according to [96], fibers of different MUs are highly intermingled. That is, in any location in the muscle, it is guaranteed that more that one MU will be represented by their muscle fibers in the recording volume of the electrode. Thus, they will contribute to the signal and will be decomposable, giving multiple sources of information on the intent.

Second, it is known [26, 27] that each MN has its own force-rate curve, i.e., the relationship between its instantaneous firing rate and the current force of muscle contraction. One can model this relationship and fit its parameters using the decomposition of contraction with a known profile (i.e, guided contraction). Then, for a new contraction, one can obtain the estimates of the intended contraction force by passing decomposed firing rates into the inverse of force-rate curves. These estimates will contain noise due to the existence of individual neural inputs to the MNs. However, supposing that these inputs are independent (that is, all the correlated part of the input is comprised in common drive), these estimates can be effectively merged into one in order to produce a global estimate of the intended contraction force.

As we have mentioned earlier, MNs of a motor neuron pool differ in sizes, which define their recruitment thresholds, force-rate curves, and contribution to the muscle contraction force. Obtaining a global estimate of the contraction force from only small MNs may not be effective at higher contraction forces since their force-rate curves may saturate [27]. On the other hand, having only large MNs , that are recruited at high contraction forces, will result in the impossibility to estimate the intent at lower ones. That is, the diversity of sizes of decomposed MNs is of interest. Experimental study [97] shows that fibers innervated by both large and small MNs

are distributed randomly in human muscles, with no tendency of forming isolated clusters of either type. Provided with that, one can count that both small and large MNs will be represented in the decomposition, which, according to the authors' experience, is usually the case in fine-wire and needle recordings.

1.5 Conclusion

Existing myoelectric control strategies utilize different modalities of EMG acquisition and of extracting intent from the signal. Most of them are based on surface recordings and feature-extraction approaches with subsequent fitting of a general machine learning model. Others attempt to establish a model-based approach, aiming to infer and interpret user's intent using knowledge about physiology and motor control. In order to access the information on neural input to a particular muscle, EMG decomposition can be used. It provides the most direct representation of the neural input, not contaminated with intent-unrelated disturbances such as electrode type or position.

This makes EMG decomposition a promising source of information in prosthetic control. However, the last requires the intent estimation to be performed in real-time, with the maximal delay of approximately 250 ms. This requirement becomes extremely challenging when considering EMG decomposition. At the moment, there are only two decomposition algorithms [91, 92] that meet it, one of which has the advantage of being adaptive to the time-varying characteristics of EMG [93].

At the current development phase, use of [92] requires intramuscular EMG recording. Above, we have listed some of the existing techniques that may provide iEMG in daily prosthetic use. The use of decomposition also brings in the requirement of high on-board computational power. As it is shown in [92], computational power is the limiting factor to the number of simultaneously decomposed MUs. This work presents a decomposition of up to 10 MUs using GPU Nvidia Tesla K80, whose dimensions make its direct application in prosthetic control impossible.

Anticipating this method's further acceleration on smaller computational powers, we address the problem of the small number of decomposed MUs in advance. Besides, other techniques of real-time EMG decomposition may be proposed in the near future, but the limitation of the small number of MUs will likely be still present. The possibility of decomposition-based prosthetic control is supported by some properties of humans' motor unit geometry and of motor control, provided in Section 1.4.3.

However, before addressing the iEMG decomposition, we will first describe our works on the sEMG. In order to achieve a better understanding of its limitations, as well as its acquisition and processing techniques, we have conducted two experimental studies. The first consisted of a classic online gesture recognition based on multichannel sEMG, feature extraction and classification. The second sough the physiological markers contained in sEMG that would permit to detect the multiple sclerosis in humans. Both studies will be presented in Chapter 2.

In the remainder of this work, we will cover the probabilistic models that incor-

porate the recruitment thresholds and force-rate curves of MNs in order to produce a global estimate of the user's intent expressed in form of contraction force or joint angle (see Chapter 3). Then, in order to study the properties of the proposed control model, we will establish a simulation of muscle contraction (see Chapter 5) and of accompanying iEMG signal (Chapter 4. Results for both simulated and experimental data will be presented in Chapter 6. Chapter 2 will describe our studies of sEMG, applied to gesture recognition and diagnostics of multiple sclerosis.

Bibliography

[1] M. Controzzi, C. Cipriani, and M. C. Carrozza, "Design of Artificial Hands: A Review," in *The Human Hand as an Inspiration for Robot Hand Development* (R. Balasubramanian and V. J. Santos, eds.), Springer Tracts in Advanced Robotics, pp. 219–246, Cham: Springer International Publishing, 2014.

[2] M. Controzzi, F. Clemente, D. Barone, A. Ghionzoli, and C. Cipriani, "The SSSA-MyHand: A dexterous lightweight myoelectric hand prosthesis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, pp. 1–1, 2016.

[3] R. Deimel and O. Brock, "A novel type of compliant and underactuated robotic hand for dexterous grasping," *The International Journal of Robotics Research*, vol. 35, pp. 161–185, Jan. 2016.

[4] J. R. Wolpaw, D. J. McFarland, G. W. Neat, and C. A. Forneris, "An EEG-based brain-computer interface for cursor control," *Electroencephalography and Clinical Neurophysiology*, vol. 78, pp. 252–259, Mar. 1991.

[5] B. Blankertz, G. Dornhege, M. Krauledat, K. R. Muller, V. Kunzmann, F. Losch, and G. Curio, "The Berlin brain-computer interface: EEG-based communication without subject training," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, pp. 147–152, June 2006.

[6] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain–computer interfaces," *Journal of Neural Engineering*, vol. 4, no. 2, p. R1, 2007.

[7] S. Waldert, "Invasive vs. Non-Invasive Neuronal Signals for Brain-Machine Interfaces: Will One Prevail?," *Frontiers in Neuroscience*, vol. 10, June 2016.

[8] E. Musk, "An integrated brain-machine interface platform with thousands of channels [White Paper]." https://www.documentcloud.org/documents/6204648-Neuralink-White-Paper.html, July 2019.

[9] E. A. Biddiss and T. T. Chau, "Upper limb prosthesis use and abandonment: A survey of the last 25 years," *Prosthetics and Orthotics International*, vol. 31, pp. 236–257, Jan. 2007.

[10] W. Schweitzer, "Technical Below Elbow Amputee Issues - Why do people stop using their prosthetic arms? [vicious cycle]," Aug. 2010.

[11] K. R. Lyons and S. S. Joshi, "Upper Limb Prosthesis Control for High-Level Amputees via Myoelectric Recognition of Leg Gestures," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, pp. 1056–1066, May 2018.

[12] T. Kuiken, "Targeted reinnervation for improved prosthetic function," *Physical Medicine and Rehabilitation Clinics of North America*, vol. 17, pp. 1–13, Feb. 2006.

[13] R. E. Burke, "Motor Units: Anatomy, Physiology, and Functional Organization," in *Comprehensive Physiology* (R. Terjung, ed.), pp. 345–422, Hoboken, NJ, USA: John Wiley & Sons, Inc., Jan. 2011.

[14] D. Stashuk, "EMG signal decomposition: How can it be accomplished and used?," *Journal of Electromyography and Kinesiology: Official Journal of the International Society of Electrophysiological Kinesiology*, vol. 11, pp. 151–173, June 2001.

[15] D. Farina, A. Crosetti, and R. Merletti, "A model for the generation of synthetic intramuscular EMG signals to test decomposition algorithms," *IEEE transactions on biomedical engineering*, vol. 48, no. 1, pp. 66–77, 2001.

[16] D. B. Sanders, K. Arimura, L. Cui, M. Ertaş, M. E. Farrugia, J. Gilchrist, J. A. Kouyoumdjian, L. Padua, M. Pitt, and E. Stålberg, "Guidelines for single fiber EMG," *Clinical Neurophysiology*, vol. 130, pp. 1417–1439, Aug. 2019.

[17] S. Muceli, W. Poppendieck, F. Negro, K. Yoshida, K. P. Hoffmann, J. E. Butler, S. C. Gandevia, and D. Farina, "Accurate and representative decoding of the neural drive to muscles in humans with multi-channel intramuscular thin-film electrodes: Multi-channel intramuscular EMG electrode," *The Journal of Physiology*, vol. 593, pp. 3789–3804, Sept. 2015.

[18] R. F. Weir, P. R. Troyk, G. A. DeMichele, D. A. Kerns, J. F. Schorsch, and H. Maas, "Implantable Myoelectric Sensors (IMESs) for Intramuscular Electromyogram Recording," *IEEE Transactions on Biomedical Engineering*, vol. 56, pp. 159–171, Jan. 2009.

[19] P. F. Pasquina, M. Evangelista, A. Carvalho, J. Lockhart, S. Griffin, G. Nanos, P. McKay, M. Hansen, D. Ipsen, J. Vandersea, J. Butkus, M. Miller, I. Murphy, and D. Hankin, "First-in-man demonstration of a fully implanted myoelectric sensors system to control an advanced electromechanical prosthetic hand," *Journal of Neuroscience Methods*, vol. 244, pp. 85–93, Apr. 2015.

[20] B. D. Farnsworth, D. M. Talyor, R. J. Triolo, and D. J. Young, "Wireless in vivo EMG sensor for intelligent prosthetic control," in *Solid-State Sensors, Actuators and Microsystems Conference, 2009*, pp. 358–361, IEEE, 2009.

[21] D. R. Merrill, J. Lockhart, P. R. Troyk, R. F. Weir, and D. L. Hankin, "Development of an Implantable Myoelectric Sensor for Advanced Prosthesis Control: Implantable Myoelectric Sensor for Advanced Prosthetic Control," *Artificial Organs*, vol. 35, pp. 249–252, Mar. 2011.

[22] S. Salminger, A. Sturma, C. Hofer, M. Evangelista, M. Perrin, K. D. Bergmeister, A. D. Roche, T. Hasenoehrl, H. Dietl, D. Farina, and O. C. Aszmann, "Long-term implant of intramuscular sensors and nerve transfers for wireless control of robotic arms in above-elbow amputees," *Science Robotics*, vol. 4, p. eaaw6306, July 2019.

[23] R. L. Drake, W. Vogl, A. W. M. Mitchell, and H. Gray, *Gray's Atlas of Anatomy - 1st Edition*. Philadelphia, PA : Churchill Livingstone/Elsevier, 3 ed., 2015.

[24] C. Heckman and R. M. Enoka, "Motor Unit," in *Comprehensive Physiology* (R. Terjung, ed.), Hoboken, NJ, USA: John Wiley & Sons, Inc., Oct. 2012.

[25] E. Henneman, G. Somjen, and D. O. Carpenter, "Functional significance of cell size in spinal motorneurons," *Journal of Neurophysiology*, vol. 28, pp. 560–580, May 1965.

[26] R. S. Person and L. P. Kudina, "Discharge frequency and discharge pattern of human motor units during voluntary contraction of muscle," *Electroencephalography and Clinical Neurophysiology*, vol. 32, pp. 471–483, May 1972.

[27] C. J. De Luca and P. Contessa, "Hierarchical control of motor units in voluntary contractions," *Journal of Neurophysiology*, vol. 107, pp. 178–195, Jan. 2012.

[28] C. J. De Luca and Z. Erim, "Common drive of motor units in regulation of muscle force," *Trends in Neurosciences*, vol. 17, pp. 299–305, Jan. 1994.

[29] S. J. Garland and T. S. Miles, "Control of motor units in human flexor digitorum profundus under different proprioceptive conditions," *The Journal of Physiology*, vol. 502, pp. 693–701, Aug. 1997.

[30] C. J. De Luca, J. A. Gonzalez-Cueto, P. Bonato, and A. Adam, "Motor Unit Recruitment and Proprioceptive Feedback Decrease the Common Drive," *Journal of Neurophysiology*, vol. 101, pp. 1620–1628, Mar. 2009.

[31] C. J. De Luca and Z. Erim, "Common Drive in Motor Units of a Synergistic Muscle Pair," *Journal of Neurophysiology*, vol. 87, pp. 2200–2204, Apr. 2002.

[32] C. J. De Luca and B. Mambrito, "Voluntary control of motor units in human antagonist muscles: Coactivation and reciprocal activation," *Journal of Neurophysiology*, vol. 58, pp. 525–542, Sept. 1987.

[33] S. Li, C. Zhuang, M. Hao, X. He, J. C. Marquez, C. M. Niu, and N. Lan, "Coordinated alpha and gamma control of muscles and spindles in movement and posture," *Frontiers in Computational Neuroscience*, vol. 9, Oct. 2015.

[34] D. Farina, F. Negro, S. Muceli, and R. M. Enoka, "Principles of Motor Unit Physiology Evolve With Advances in Technology," *Physiology*, vol. 31, pp. 83–94, Mar. 2016.

[35] A. M. Castronovo, F. Negro, S. Conforto, and D. Farina, "The proportion of common synaptic input to motor neurons increases with an increase in net excitatory input," *Journal of Applied Physiology*, vol. 119, pp. 1337–1346, Dec. 2015.

[36] K. C. McGill, Z. C. Lateva, and H. R. Marateb, "EMGLAB: An interactive EMG decomposition program," *Journal of Neuroscience Methods*, vol. 149, pp. 121–133, Dec. 2005.

[37] E. M. Schmidt, "Unit activity from peripheral nerve bundles utilizing correlation techniques," *Medical and biological engineering*, vol. 9, pp. 665–674, Nov. 1971.

[38] G. L. Gerstein and W. A. Clark, "Simultaneous Studies of Firing Patterns in Several Neurons," *Science*, vol. 143, pp. 1325–1327, Mar. 1964.

[39] R. S. LeFever and C. J. De Luca, "A procedure for decomposing the myoelectric signal into its constituent action potentials-part I: Technique, theory, and implementation," *IEEE transactions on biomedical engineering*, no. 3, pp. 149–157, 1982.

[40] K. C. McGill, K. L. Cummins, and L. J. Dorfman, "Automatic decomposition of the clinical electromyogram," *IEEE Transactions on Biomedical Engineering*, no. 7, pp. 470–477, 1985.

[41] D. Stashuk and Y. Qu, "Adaptive motor unit action potential clustering using shape and temporal information," *Medical and Biological Engineering and Computing*, vol. 34, no. 1, pp. 41–49, 1996.

[42] J. Florestal, P. Mathieu, and A. Malanda, "Automated decomposition of intramuscular electromyographic signals," *IEEE Transactions on Biomedical Engineering*, vol. 53, pp. 832–839, May 2006.

[43] H. R. Marateb, S. Muceli, K. C. McGill, R. Merletti, and D. Farina, "Robust decomposition of single-channel intramuscular EMG signals at low force levels," *Journal of Neural Engineering*, vol. 8, p. 066015, Oct. 2011.

[44] M. H. Hassoun, C. Wang, and A. R. Spitzer, "NNERVE: Neural network extraction of repetitive vectors for electromyography. I. Algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 41, no. 11, pp. 1039–1052, 1994.

[45] J. Florestal, P. Mathieu, and R. Plamondon, "A Genetic Algorithm for the Resolution of Superimposed Motor Unit Action Potentials," *IEEE Transactions on Biomedical Engineering*, vol. 54, pp. 2163–2171, Dec. 2007.

[46] R. Gut and G. S. Moschytz, "High-precision EMG signal decomposition using communication techniques," *IEEE transactions on signal processing*, vol. 48, no. 9, pp. 2487–2494, 2000.

[47] H. G. Rey, C. Pedreira, and R. Quian Quiroga, "Past, present and future of spike sorting techniques," *Brain Research Bulletin*, vol. 119, pp. 106–117, Oct. 2015.

[48] P. C. Petrantonakis and P. Poirazi, "A Simple Method to Simultaneously Detect and Identify Spikes from Raw Extracellular Recordings," *Frontiers in Neuroscience*, vol. 9, Dec. 2015.

[49] R. Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural computation*, vol. 16, no. 8, pp. 1661–1687, 2004.

[50] A. Holobar and D. Zazula, "Multichannel Blind Source Separation Using Convolution Kernel Compensation," *IEEE Transactions on Signal Processing*, vol. 55, pp. 4487–4496, Sept. 2007.

[51] A. Holobar, D. Farina, M. Gazzoni, R. Merletti, and D. Zazula, "Estimating motor unit discharge patterns from high-density surface electromyogram," *Clinical Neurophysiology*, vol. 120, pp. 551–562, Mar. 2009.

[52] F. Negro, S. Muceli, A. M. Castronovo, A. Holobar, and D. Farina, "Multi-channel intramuscular and surface EMG decomposition by convolutive blind source separation," *Journal of Neural Engineering*, vol. 13, p. 026027, Feb. 2016.

[53] V. Glaser and A. Holobar, "Motor Unit Identification From High-Density Surface Electromyograms in Repeated Dynamic Muscle Contractions," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, pp. 66–75, Jan. 2019.

[54] C. J. De Luca, "Decomposition of Surface EMG Signals," *Journal of Neurophysiology*, vol. 96, pp. 1646–1657, Jan. 2006.

[55] Y. Ning, Y. Zhao, A. Juraboev, P. Tan, J. Ding, and J. He, "Multichannel Surface EMG Decomposition Based on Measurement Correlation and LMMSE," *Journal of Healthcare Engineering*, vol. 2018, pp. 1–12, June 2018.

[56] A. Fougner, O. Stavdahl, P. J. Kyberd, Y. G. Losier, and P. A. Parker, "Control of Upper Limb Prostheses: Terminology and Proportional Myoelectric Control - A Review," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, pp. 663–677, Sept. 2012.

[57] R. Reiter, "Eine neue Elektrokunsthand," *Grenzgebiete Der Medizin*, vol. 1, pp. 133–135, Sept. 1948.

[58] K. J. Zuo and J. L. Olson, "The evolution of functional hand replacement: From iron prostheses to hand transplantation," *Plastic Surgery*, vol. 21, no. 1, 2014.

[59] A. Kobrinski, S. Bolkovitin, L. Voskoboinikova, D. Ioffe, E. Polyan, B. Popov, Y. Slavutski, A. Sysin, and Y. Yakobson, "Problems of bioelectric control," *Automatic and Remote Control, Proc. 1st IFAC Int. Congress*, vol. 2, p. 619, 1960.

[60] E. D. Sherman, "A Russian Bioeleciric-Controlled Prosthesis: Report of a Research Team from the Rehabilitation Institute of Montreal," *Canadian Medical Association Journal*, vol. 91, p. 3, Dec. 1964.

[61] "Below-elbow prosthesis with Michelangelo — Ottobock UK." https://www.ottobock.co.uk/prosthetics/upper_limbs_prosthetics/product-systems/below_elbow_prosthesis_featuring_michelangelo/.

[62] "How the i-limb works — Touch Bionics." https://www.touchbionics.com/products/how-i-limb-works.

[63] K. Veer and T. Sharma, "A novel feature extraction for robust EMG pattern recognition," *Journal of Medical Engineering & Technology*, vol. 40, pp. 149–154, May 2016.

[64] M. Hakonen, H. Piitulainen, and A. Visala, "Current state of digital signal processing in myoelectric interfaces and related applications," *Biomedical Signal Processing and Control*, vol. 18, pp. 334–359, Apr. 2015.

[65] K. Xing, P. Yang, J. Huang, Y. Wang, and Q. Zhu, "A real-time EMG pattern recognition method for virtual myoelectric hand control," *Neurocomputing*, vol. 136, pp. 345–355, July 2014.

[66] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Systems with Applications*, vol. 40, pp. 4832–4840, Sept. 2013.

[67] J.-U. Chu, I. Moon, Y.-J. Lee, S.-K. Kim, and M.-S. Mun, "A Supervised Feature-Projection-Based Real-Time EMG Pattern Recognition for Multifunction Myoelectric Hand Control," *IEEE/ASME Transactions on Mechatronics*, vol. 12, pp. 282–290, June 2007.

[68] N. Jarrassé, E. de Montalivet, F. Richer, C. Nicol, A. Touillet, N. Martinet, J. Paysant, and J. B. de Graaf, "Phantom-Mobility-Based Prosthesis Control in Transhumeral Amputees Without Surgical Reinnervation: A Preliminary Study," *Frontiers in Bioengineering and Biotechnology*, vol. 6, p. 164, Nov. 2018.

[69] Jun-Uk Chu, Inhyuk Moon, and Mu-Seong Mun, "A Real-Time EMG Pattern Recognition System Based on Linear-Nonlinear Feature Projection for a Multifunction Myoelectric Hand," *IEEE Transactions on Biomedical Engineering*, vol. 53, pp. 2232–2239, Nov. 2006.

[70] K. Akhmadeev, E. Rampone, T. Yu, Y. Aoustin, and E. Le Carpentier, "A testing system for a real-time gesture classification using surface EMG," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 11498–11503, 2017.

[71] D. Tkach, H. Huang, and T. A. Kuiken, "Study of stability of time-domain features for electromyographic pattern recognition," *Journal of NeuroEngineering and Rehabilitation*, vol. 7, p. 21, May 2010.

[72] E. Scheme and K. Englehart, "On the robustness of EMG features for pattern recognition based myoelectric control; A multi-dataset comparison," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 650–653, Aug. 2014.

[73] "Coapt." http://www.coaptengineering.com/.

[74] "Bebionic hand." https://www.ottobock.co.uk/prosthetics /upper_limbs_prosthetics.

[75] W. Schweitzer, "Technical Below Elbow Amputee Issues - The "Russian arm" control paradigm for myoelectric prostheses [still relabeled as "bionic" or "thought controlled" arm] what we have been withholding from you so far," June 2014.

[76] Ning Jiang, K. Englehart, and P. Parker, "Extracting Simultaneous and Proportional Neural Control Information for Multiple-DOF Prostheses From the Surface Electromyographic Signal," *IEEE Transactions on Biomedical Engineering*, vol. 56, pp. 1070–1080, Apr. 2009.

[77] S. Muceli, N. Jiang, and D. Farina, "Extracting Signals Robust to Electrode Number and Shift for Online Simultaneous and Proportional Myoelectric Control by Factorization Algorithms," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, pp. 623–633, May 2014.

[78] A. Stango, F. Negro, and D. Farina, "Spatial Correlation of High Density EMG Signals Provides Features Robust to Electrode Number and Shift in Pattern Recognition for Myocontrol," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, pp. 189–198, Mar. 2015.

[79] M. Sartori, G. Durandau, S. Došen, and D. Farina, "Robust simultaneous myoelectric control of multiple degrees of freedom in wrist-hand prostheses by real-time neuromusculoskeletal modeling," *Journal of Neural Engineering*, vol. 15, p. 066026, Dec. 2018.

[80] C. Cipriani, J. L. Segil, J. A. Birdwell, and R. F. ff Weir, "Dexterous Control of a Prosthetic Hand Using Fine-Wire Intramuscular Electrodes in Targeted Extrinsic Muscles," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, pp. 828–836, July 2014.

[81] L. H. Smith, T. A. Kuiken, and L. J. Hargrove, "Real-time simultaneous and proportional myoelectric control using intramuscular EMG," *Journal of Neural Engineering*, vol. 11, p. 066013, Dec. 2014.

[82] E. N. Kamavuako, E. J. Scheme, and K. B. Englehart, "Wrist torque estimation during simultaneous and continuously changing movements: Surface vs. untargeted intramuscular EMG," *Journal of Neurophysiology*, vol. 109, pp. 2658–2665, Mar. 2013.

[83] E. N. Kamavuako, K. B. Englehart, W. Jensen, and D. Farina, "Simultaneous and Proportional Force Estimation in Multiple Degrees of Freedom From Intramuscular EMG," *IEEE Transactions on Biomedical Engineering*, vol. 59, pp. 1804–1807, July 2012.

[84] D. Y. Barsakcioglu and D. Farina, "A real-time surface EMG decomposition system for non-invasive human-machine interfaces," in *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, (Cleveland, OH), pp. 1–4, IEEE, Oct. 2018.

[85] A. Holobar and D. Farina, "Blind source identification from the multichannel surface electromyogram," *Physiological Measurement*, vol. 35, pp. R143–R165, July 2014.

[86] D. Farina, I. Vujaklija, M. Sartori, T. Kapelner, F. Negro, N. Jiang, K. Bergmeister, A. Andalib, J. Principe, and O. C. Aszmann, "Man/machine interface based on the discharge timings of spinal motor neurons after targeted muscle reinnervation," *Nature Biomedical Engineering*, vol. 1, p. 0025, Feb. 2017.

[87] T. Kapelner, I. Vujaklija, N. Jiang, F. Negro, O. C. Aszmann, J. Principe, and D. Farina, "Predicting wrist kinematics from motor unit discharge timings for the control of active prostheses," *Journal of NeuroEngineering and Rehabilitation*, vol. 16, p. 47, Apr. 2019.

[88] C. K. Thompson, F. Negro, M. D. Johnson, M. R. Holmes, L. M. McPherson, R. K. Powers, D. Farina, and C. J. Heckman, "Robust and accurate decoding of motoneuron behaviour and prediction of the resulting force output: Neural drive to muscle and resulting force output," *The Journal of Physiology*, vol. 596, pp. 2643–2659, July 2018.

[89] Y. Li, L. H. Smith, L. J. Hargrove, D. J. Weber, and G. E. Loeb, "Sparse Optimal Motor Estimation (SOME) for Extracting Commands for Prosthetic

Limbs," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, pp. 104–111, Jan. 2013.

[90] L. H. Smith, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "Determining the Optimal Window Length for Pattern Recognition-Based Myoelectric Control: Balancing the Competing Effects of Classification Error and Controller Delay," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, pp. 186–192, Apr. 2011.

[91] V. Glaser, A. Holobar, and D. Zazula, "Real-Time Motor Unit Identification From High-Density Surface EMG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, pp. 949–958, Nov. 2013.

[92] T. Yu, K. Akhmadeev, E. L. Carpentier, Y. Aoustin, and D. Farina, "On-line recursive decomposition of intramuscular EMG signals using GPU-implemented Bayesian filtering," *IEEE Transactions on Biomedical Engineering*, 2019. Manuscript in revision.

[93] T. Yu, K. Akhmadeev, E. L. Carpentier, Y. Aoustin, R. Gross, Y. Pereon, and D. Farina, "Recursive decomposition of electromyographic signals with a varying number of active sources: Bayesian modelling and filtering," *IEEE Transactions on Biomedical Engineering*, pp. 1–1, 2019.

[94] J. Monsifrot, E. Le Carpentier, Y. Aoustin, and D. Farina, "Sequential Decoding of Intramuscular EMG Signals via Estimation of a Markov Model," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, pp. 1030–1040, Sept. 2014.

[95] B. Feinstein, B. Lindegård, E. Nyman, and G. Wohlfart, "Morphologic studies of motor units in human muscles," *Cells Tissues Organs*, vol. 23, no. 2, pp. 127–142, 1955.

[96] F. Buchthal, C. Guld, and P. Rosenfalck, "Multielectrode Study of the Territory of a Motor Unit," *Acta Physiologica Scandinavica*, vol. 39, pp. 83–104, Jan. 1957.

[97] M. A. Johnson, J. Polgar, D. Weightman, and D. Appleton, "Data on the distribution of fibre types in thirty-six human muscles: An autopsy study," *Journal of the Neurological Sciences*, vol. 18, pp. 111–129, Jan. 1973.

Chapter 2

Surface EMG studies. Gesture classification and multiple sclerosis diagnosis.

2.1 Introduction

This chapter describes two studies on sEMG that we have conducted during this thesis. First study realizes a classic sEMG pattern recognition approach to the intended gesture inference and tests its precision. The second seeks the EMG-specific physiological markers related to the presence and progression of multiple sclerosis (MS) in humans.

In both studies, we used MYO armband by Thalmic Labs Inc as a signal acquisition system. This chapter will start with the description of the armband's characteristics, its strong and weak points. Next, we will separately describe each study in Sections 2.2 and 2.3. Conclusions on the two projects, as well as general conclusion on the device, will be given at the end of the chapter.

MYO™ armband is a device developed by Thalmic Labs Inc.™. It integrates eight surface EMG (sEMG) sensors (see Figure 2.1) and an inertial measurement unit (IMU). sEMG signal is sampled with eight bit precision at frequency of 200 Hz and is wirelessly transmitted to a PC using Bluetooth Low Energy (BLE) protocol. Its default software is capable of recognizing five different hand gestures based on EMG, while IMU sensor provides pointer control. In academic studies, this device was applied in prosthetic control [1], sign language gesture recognition [2] and control of robotic manipulators [3], [4].

The first study addresses the development of a testing system for pattern-recognition-based strategies of myoelectric control. The following section describes the structure and components of the proposed system, as well as a process of its testing. The last included an acquisition of an accompanying EMG, using Myo™ armband by Thalmic Labs Inc.™, for six different gestures (classes) from seven subjects, as well as its processing, feature extraction, training the classifier and further real-time validation. The results show that system provides acceptable classification rates.

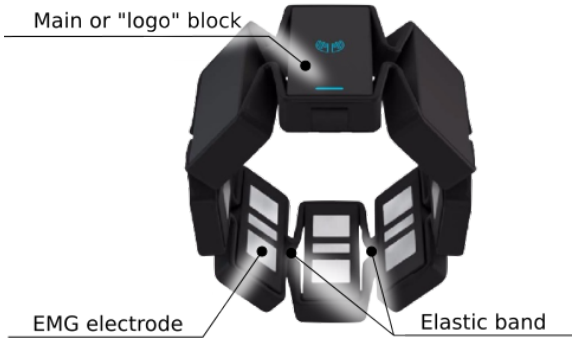


Figure 2.1: MYO armband by Thalmic Labs Inc., adapted from from the official website

.

The second study used the same device to acquire EMG and IMU data from both multiple sclerosis and control subjects in order to provide a diagnostic tool based on EMG features. The Multiple sclerosis (MS) is a major auto-immune disease that is the leading cause of non-traumatic impairment of the central nervous system (CNS) in young adults. Successful treatment of MS patients depends on accurate tools for both the MS diagnosis and the disability progression. In current and upcoming studies the authors aim to explore the capabilities of applying a commercial electromyographic and inertial sensor, coupled with a multichannel signal processing tool, to standard neurological examination of MS progression. In this pilot study we formulate a two-class "healthy control" - "having MS" classification problem. A dataset of electromyographic signals and inertial sensor measurements from 71 individuals (31 MS patients and 40 healthy controls) was acquired during standard neurological examination routine. Temporal and spectral features of the signals were extracted in order to train and validate a classification model. Finally, a Support Vector Machine classifier was obtained giving AUROC = 0.94, 95% CI = [0.88, 0.99]. We propose a set of signal descriptors that correlate with the objective components of the neurological examination. The proposed signal acquisition and processing technique, being easy to integrate into the traditional neurological exam, may have high potential for aiding in quantifying MS progression.

2.2 Real-time gesture classification using commercial EMG armband[1]

2.2.1 Introduction

Myoelectric control, i.e. derivation of user's intent from accompanying electromyogram (EMG) emerges fast during last three decades. It finds its applications in human-machine interfaces, such as virtual reality ([6]), electrical wheelchairs ([7]),

[1]This section is based on the authors' paper [5]

exoskeletons ([8]) etc. Another challenging application of myoelectric control is prosthetics, especially hand prosthetics due to its importance for amputees' rehabilitation and complexity of the problem.

The most of the existing intent derivation algorithms are based on pattern recognition. In this approach, the set of all possible hand gestures (including its positions and velocities) is separated to form a number of discrete subsets, or classes ([9, 10]). These subsets are chosen to cover gestures that are considered to be the most important and, at the same time, that they would be distinguishable by a pattern recognition algorithm with a given accuracy. The most frequently accepted classes are cylindrical grasp, hand rotations, wrist flexions and extensions, radial and ulnar deviations ([11, 12]).

To apply classification techniques to EMG one should first reduce its dimensionality. This is usually achieved by means of feature extraction. Features are scalars or vectors that represent temporal or frequency characteristics of the signal. Number of their types presented in the literature is large ([13, 10]), and it is shown that some of them are more effective that others ([14, 13]).

Once features are extracted, they are arranged to form a so-called feature vector. This vector is then used as an input for a classifier. Classifiers mostly used in related studies are linear discriminant analysis ([15]), neural networks ([16]) and support vector machines ([17]).

Performance of pattern-recognition-based control algorithms presented in the literature is usually higher than 98% (in terms of correct classification rate) even for large sets of classes. But despite all the advances the application of this approach to human-machine interfaces is developing slowly, especially when commercial prostheses are considered. There are several reasons for this situation ([18]), among which are lack of adaptivity and robustness, absence of feedback and others. Finally, the problem is aggravated by high costs of myoelectric prostheses which varies between 20.000 and 100.000$ [19].

The interest of this study was to develop a testing system for pattern-recognition-based control strategies with a possibility of real-time operation. Generally, such systems include EMG acquisition and data transmission hardware, a toolbox for feature extraction and signal processing functions, an interface that automates the test, and configurable classifier. To test the presented system, an experimental setup was developed including signal acquisition, training and testing procedures, subject position, etc. Details of realization are covered in the next section.

Several analogous systems are presented in the literature, mostly concerning custom-made acquisition systems, which opens way to higher sampling rates and problem-oriented design. In some systems such advantages lead to higher number of classified gestures ([20]) or reduced number of channels ([21, 22]). But at the same time a custom design is difficult to reproduce only by following the related paper. In contrary, this study is based on an acquisition system that has a generic design and is easily accessible in many countries. System presented in ([23]) meets this demand although it lacks a clear step-by-step explanation of data processing part.

The remainder of this section is organized as follows. Section 2.2.2 introduces the equipment and software used in the study. Section 2.2.3 describes signal process-

ing and classification methods. Section 2.2.5 presents several experimental results. Section 2.2.6 offers conclusions and suggestions about future work.

2.2.2 Experimental equipment and protocol

Equipment

In this study, to acquire electromyogram (EMG), we used MYO™ armband. To collect, process and manage the database of received sEMG, an environment was created on a base of MATLAB™. It consists of a graphical user interface and tools for database management, signal processing and experiment planning.

To transfer data from official standard Myo™ armband software Myo Connect to MATLAB™, the Myo SDK MATLAB MEX Wrapper ([24]) was used. It is a library that converts original C++ functions of MYO SDK into MEX files callable from MATLAB™.

Experimental Protocol

This study involved seven normally limbed right-handed subjects: two females and five males all are $23-28$ years old. One of them was familiar with the experimental setup while the others were inexperienced.

During the experiment, each subject sit on a chair wearing the armband on the right forearm. During signal acquisition, subjects were instructed to keep the angle at the elbow joint at approximately 90°. Between sessions, the subject could hang the hand down or put the elbow carefully on the table avoiding shifting of the armband.

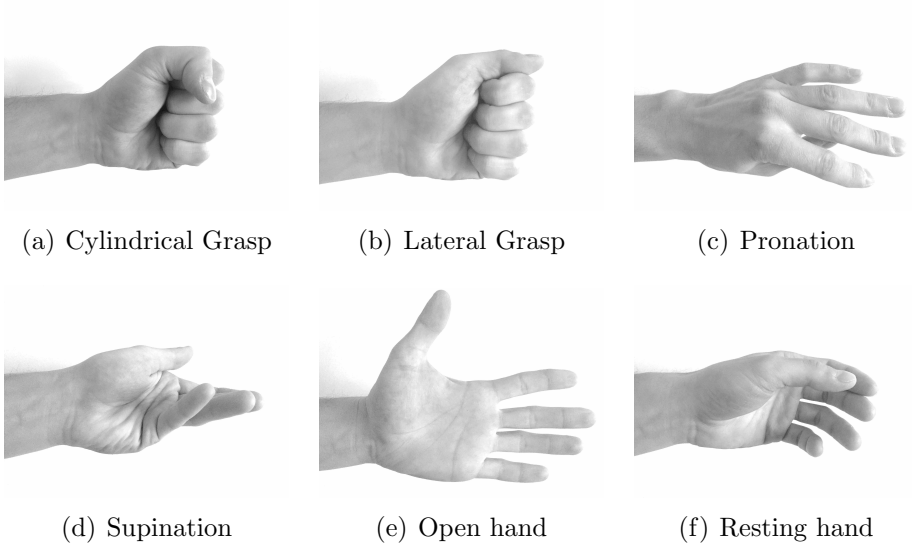
The armband was placed around the proximal portion of the forearm. Specifically, the lateral side of the armband coincided approximately with the middle of the radius bone. The plane passing through centers of all electrodes of the armband was perpendicular to the forearm and electrodes were aligned with each other. At the beginning of the experiment the position of the armband was contoured by a thin black marker, in order to keep trace of its starting position.

Five hand gestures were chosen for classification in this study (Fig. 2.2): cylindrical grasp, lateral grasp, spread fingers, pronation and supination plus resting position, having six motion classes in total.

The experiment consisted of three major phases. The first one was acquiring training set and training the classifier. The second phase included the online validation of the gesture classification. The third one consisted of taking the armband off the hand and putting it back on its position, using previously drawn contours, and repeating the second phase.

As was already mentioned, a software based on MATLAB™ was developed to maintain the experimental process. It automatically instructed the subject when and which gesture to perform.

The training set acquisition was divided in several similar trials, each containing short sessions (about ten seconds each) of each gesture execution. Signal acquisition always started three seconds after the subject reached target position of the hand.



(a) Cylindrical Grasp (b) Lateral Grasp (c) Pronation

(d) Supination (e) Open hand (f) Resting hand

Figure 2.2: Grasp types or classes used in this study.

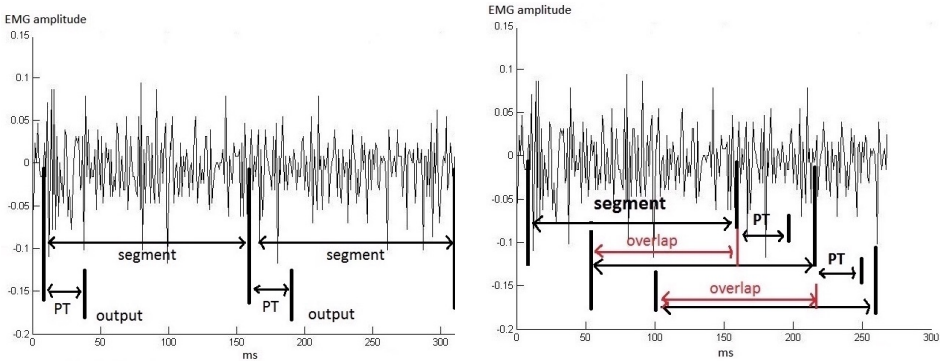
That ensures the acquired EMG be in steady-state. Subjects were also instructed to maintain a medium force of contraction, but no constraints were set on it and no monitoring had place.

During the online validation part, sEMG signal was processed and classification was done in real-time with constant delay of about 250 ms which is considered acceptable in this type of prosthetic control ([25]). It is important regarding the fact that this system is to be used later in real-time control of a robotic hand. Results of online classification were provided to the subject while he was executing the gestures.

The overall duration of the experiment was about 35 minutes which included ten minutes for data acquisition, five minutes for classifier training, five minutes for each online validation and ten minutes for the equipment setup and instructing the subject.

2.2.3 Feature extraction

In order to use EMG in classification, one should first compress it to reduce its dimensionality and discard irrelevant information. This process is usually referred as feature extraction. Many types of EMG features are presented in the literature ([10]), and for this study computationally the most simple and often used ones were chosen ([26]): mean absolute value (MAV), zero crossings (ZC), slope sign changes (SSL), waveform length (WL) and autoregressive coefficients (AR). All these features except AR are time-domain features, whereas AR was added to represent spectral characteristics of the signal. All these features are calculated as follows ([27]).



(a) Signal windowing without overlapping

(b) Signal windowing with overlapping.

Figure 2.3: Two types of signal windowing: (a) - without overlapping, (b) - with overlapping. PT stands for processing time for a signal segment within one window.

Mean Absolute Value.

The Mean Absolute Value (MAV) feature provides an estimate of mean energy of the signal x in a window containing N samples. It can be formulated as:

$$\text{MAV} = \frac{1}{N} \sum_{k=1}^{N} |x_k| \tag{2.1}$$

where:

- x_k is the k-th sample in the window.

Waveform Length.

Waveform Length (WL) feature is the cumulative length of the signal waveform in a window of N samples. Its formula is:

$$\text{WL} = \sum_{k=2}^{N} |\Delta x_k| \quad \text{where} \quad \Delta x_k = x_k - x_{k-1} \tag{2.2}$$

It characterizes the complexity of signal waveform in a window.

Zero Crossing.

The Zero Crossing (ZC) feature counts the number of times the waveform passes by zero. In practice, a threshold must be applied to this feature in order to suppress noise-induces crossings. Considering threshold T, a signal window containing N samples, and taking two consecutive samples x_k and x_{k+1}, there is a zero crossing between the two if the following condition is satisfied:

$$g_{\mathrm{ZC}}(x_k) = \begin{cases} 1 & \text{if } x_k x_{k+1} < 0 \text{ and } |x_k - x_{k+1}| \geq T \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

For this study a threshold of 3% of maximum signal level was applied. This level was defined by the acquisition equipement. Thus, the ZC feature is defined as:

$$\mathrm{ZC} = \sum_{k=1}^{N-1} g_{\mathrm{ZC}}(x_k) \tag{2.4}$$

This feature summarises frequency-related properties of the signal.

Slope Sign Change.

The Slope Sign Change (SSC) feature counts the number of times the signal slope changes its sign. The same threshold as for zero-crossings feature was applied to suppress noise. Given threshold T, three consecutive samples x_{k-1}, x_k, and x_{k+1} of a signal segment with N samples, change of slope sign condition is:

$$(x_k - x_{k-1})\,(x_k - x_{k+1}) \geq T \tag{2.5}$$

Autoregressive Coefficients.

The Autoregressive (AR) process models time-varying stochastic processes, like EMG signals, as auto-regression time series. The output variable of the AR model is a linear combination of its own previous values plus gaussian white noise. It can be defined as:

$$x_k = \sum_{i=1}^{s} a_i\, x_{k-i} + w_k \tag{2.6}$$

where:

- a_i are the autoregressive coefficients;
- s is the order of the AR model;
- w_k is the residual term (gaussian white noise).

Autoregressive coefficients can be estimated by fitting the AR model to the time series using least squares procedure or method of moments (Yule-Walker equations).

Window overlapping

Since the processing time in online validation was found to be less than accepted processing delay, an overlapped windowing was applied to the signals. Overlapping permits to produce more classifications during the chosen processing delay and then, applying a majority voting to them, increase the stability and accuracy of control ([10]). The principle of overlapped windowing is illustrated on Fig. 2.3. For this

study, processing delay of 250 ms and overlap of 125 ms were chosen, making the system produce classification results each 125 ms.

Features listed above were extracted from each signal segment, which produced a vector of ten elements: four scalars features (MAV, ZC, SSL, WL) plus six AR coefficients. Having eight sEMG channels, one obtains a global vector of 80 features.

To reduce such a high dimensionality of feature vector, principle component analysis (PCA) was applied. It is a common technique which consists of converting the initial dataset into a set of uncorrelated values using orthogonal transformation. As a result, nine principal components representing 97% of total variance were extracted. These components were mostly spanned by WL (waveform length), ZC (zero crossing) and SSC (slope sign changes) and only four first AR coefficients were contributing in this set. Obtained principal components were used to project feature vectors and decrease their dimensionality from 80 to nine.

2.2.4 Gesture classification

Choice of the classification model

As a classifier model for this study, support vector machine (SVM) with round-based kernel function (RBF) was applied. SVMs are supervised learning models that separate training samples into two classes in a way that the marge between them is maximized. Associated problem is formulated as a regularized quadratic programming with linear inequality constraints for which there are effective methods for finding solution. When classification problem is highly non-linear, so-called kernel trick is used which is an implicit projection of training dataset to higher-dimensional space where the classes may have better separability. Since SVMs use matrix of pairwise scalar products instead of the dataset itself, replacing the standard scalar product by a kernel function implicitly projects the data to other spaces. Our choice of round-based kernel functions is the result of preliminary testing in which we found this type of kernel to be the most effective one. One-versus-one approach was adopted to establish a multi-class classification. A MATLAB™ version of LIBSVM (Library for Support Vector Machines, see [28]) was used to implement this classifier.

Training procedure

Before passing to the classifier, the feature vectors were scaled to the range $[0, 1]$. The main advantage of data scaling is avoiding attributes having greater numeric ranges dominate those in smaller numeric ranges ([28]). The scaling factors were saved in order to apply them to the testing dataset.

To train an SVM classifier, one should also find best values of SVM regularization parameter C and of kernel parameter γ. Since optimal values of these parameters may differ from subject to subject it is necessary to estimate them for each experiment. For that, a grid search was implemented, which is a common technique in case of SVM classifier training. It consists of, having a discrete set of possible values for each unknown parameter, repeating the training process for all their possible

combinations and then choosing the best one. Also cross-validation process was automatically applied by LIBSVM during its training procedure.

Majority voting

During the online validation of gesture classification, final decision about class label was corrected by majority voting. According to ([29]) majority voting considers the most recent $2m+1$ estimated classes (m-estimations before and m-estimations after a considered estimation). Then it outputs the value that occurs most as correct estimation. The majority voting technique introduces a delay in the generation of the output, because it waits for the next m-estimations before producing an output. As was already mentioned, processing delay was chosen to be equal to 250 ms. Hence, real-time constraints impose ([29]): In this study $m = 2$ was chosen, that implies an overlap of 125 ms.

All resulting class labels were collected and used later to compute confusion matrices presented in the next section.

2.2.5 Results

Classifier training

Table 2.1: Cross-Validation accuracy for all subjects and classes across all trials.

	Cross Validation accuracy of each motion class (%)						**Mean**
	Cylindr. grasp	Open	Lateral grasp	Pron.	Supin.	Rest	
Sub 1	100	100	100	98.9	100	100	99.81
Sub 2	96.6	100	94.4	99.4	98.9	97.2	97.75
Sub 3	93.3	100	97.8	93.3	100	100	97.4
Sub 4	98.9	100	97.8	100	100	100	99.45
Sub 5	98.9	100	100	98.9	100	98.9	99.45
Sub 6	100	100	100	100	100	100	100
Sub 7	99.4	100	98.3	99.4	98.9	100	99.3
Mean	98.16	100	98.3	98.56	99.68	99.44	**99.023**

Summary of training procedure results is presented in Table 2.1. It contains rates (in percents) of correct classifications for given class and given subject, as well as corresponding mean values. Confusion matrices for the best and worst subjects (in the sense of classification results) are presented in Tables 2.2 and 2.3, CG stands for cylindrical grasp, O for open hand, LG for lateral grasp, P for pronation, S for supination and R for rest. Pairs of classes that produced the most errors across all subjects were Lateral Grasp - Cylindrical Grasp and Pronation - Cylindrical Grasp.

Table 2.2: Confusion matrix for user with best performance

Best subject						
	CG	**O**	**LG**	**P**	**S**	**R**
CG	180	0	0	0	0	0
O	0	180	0	0	0	0
LG	0	0	180	0	0	0
P	0	0	2	178	0	0
S	0	0	0	0	180	0
R	0	0	0	0	0	180

Table 2.3: Confusion matrix for user with worst performance

Worst subject						
	CG	**O**	**LG**	**P**	**S**	**R**
CG	168	0	0	12	0	0
O	0	180	0	0	0	0
LG	0	0	176	4	0	0
P	10	0	2	168	0	0
S	0	0	0	0	180	0
R	0	0	0	0	0	180

Online validation

For each class, three classification rates were calculated: the one of the training procedure, and two obtained from online validation before and after taking-off and putting on the bracelet. All three, averaged across subjects, are shown on figure 2.4. One can see that classification rate stays at considerably high level after the removal and putting on the armband.

2.2.6 Conclusion

Testing system established in this study is simple and can be easily reproduced by anyone since it is based on widely distributed hardware (Myo™ armband by Thalmic Labs Inc.™ and an ordinary PC with Windows 7™ system), open source libraries (LIBSVM, MYO MEX Wrapper) and MATLAB™.

We note that presented system supports real-time intent recognition and provides it for current choice of features and classifier. Authors consider results of training procedure and of online validation to be acceptable [1]. The main reason of this study was to assess the performance of the developed system, as well as performance of its components, and this task is considered to be done.

System developed in this study can serve for testing new classification algorithms,

[1] Video example available under this link

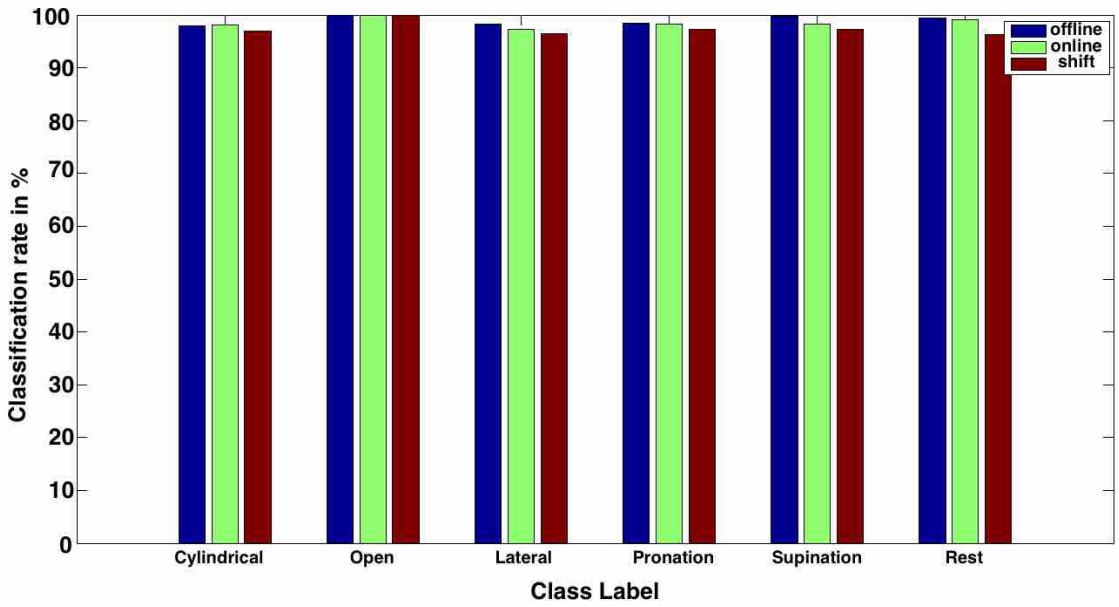


Figure 2.4: Accuracies of, respectively: training ("offline" in legend), online validation ("online") and online validation after taking off and putting the armband on ("shift").

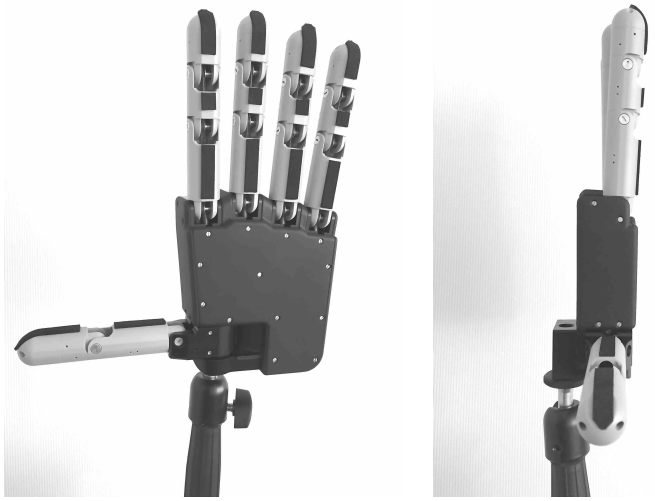


Figure 2.5: Front and side views of the robotic hand used in the tests.

experiment protocols and signal features. It can also be used in benchmarking for analysis of other algorithms of myoelectric control. Another way to use such a system is real-time control of a robotic hand. We have already conducted[1] preliminary tests with one such hand (see Fig. 2.5). Following tests may include assessment of applicability of existing control algorithms as well as usability of such kind of

[1]Video example available under this link

manipulators.

Gesture recognition based on sEMG can be straightforwardly implemented and provide promising results in a laboratory environment. We did not aim to conduct a full analysis of its applicability in the prosthetic control, including such factors posture influence, sweat, skin state after a long-term use, and user' fatigue. These issues, however, are critical and make the control highly unreliable even in the much simpler two-channel strategy. As we have mentioned in the Introduction (see Section 1.2.5), these factors can be partly alleviated by using the intramuscular EMG and EMG decomposition. A strategy of user intent derivation from EMG decomposition will be described in Chapter 3.

2.3 Aid in multiple sclerosis diagnosis using commercial EMG armband[1]

2.3.1 Introduction

Multiple sclerosis (MS) is a chronic debilitating neurological disorder that mainly affects young individuals aged between 20 and 40 years old. As a cause of neurologic disability MS is second only to trauma, having its prevalence estimated at 2.5 million worldwide in 2014. The actual cause of MS is yet to be identified, but a complex interaction between genetic and environmental factors contributes to the risk. To date, there is no reliable method to predict MS onset or progression. Successful managing of the symptoms and attacks for MS patients highly depends on an accurate and timely diagnosis as well as the possibility to measure disability progression.

Diagnostic criteria for multiple sclerosis include a number of clinical and paraclinical laboratory assessments [31, 32]: cerebrospinal fluid analysis, study of visual evoked potentials, electromyography analysis, neuroimaging and motor/sensory/balance function tests. The latter involves various motor tasks to be accomplished by the subject: timed 25-foot walk [33], 9-hole peg test, finger-to-nose test [34], heel-knee-shin test, finger tapping, foot tapping, etc.

The most common motor manifestations of MS are muscle fatigue, spasticity and tremor. Listed symptoms involve abnormal functioning of skeletal muscles and thus affect their activation patterns. In such cases, deviations may be revealed by analysis of limb trajectories and of involved muscles' electromyography (EMG). These measurements are proven to be efficient in different studies of MS progression [35, 36, 37]. Thus, an EMG recording along with the inertial measurement unit (IMU) data may aid to characterise presence and severity of MS.

Common MS diagnosis and progression study approaches, as those listed above, require specific equipment, procedures and clinical expertise. A lack of them may slow down the diagnosis and evaluation of MS progression or even make it impossible, which is a common case for low populated areas or developing countries. A possible

[1]This section is based on the authors' paper [30]

way to overcome these difficulties is to apply a widespread cheap acquisition system, along with unified assessment protocol and automated decision-making. As such an acquisition system we propose the MYO armband.

Typical signal processing pipeline in these applications consists of the following steps: windowing, feature extraction, dimensionality reduction and classification using machine learning techniques [38, 39]. Such an approach may also be effective in an application to MS diagnostics since there is no strictly defined model of how MS affects surface EMG signals or limb trajectories. Other reasons to use machine learning techniques in this case are the dimensionality of the data and the fact that measurements are to be made with two different kinds of sensors (EMG and IMU).

In the sequel, we present the entire processing pipeline including the data acquisition, signal preprocessing, dimensionality reduction and decision-making steps. At last, experimental results involving MS patients and healthy controls are discussed.

In this study we focus on a basic two-class (healthy controls vs. diagnosed MS) classification problem to investigate the discriminative capabilities of measurements provided by aforementioned device. Solving such a problem is not intended to bring any benefits compared to doctor examination, but instead is considered only as proof-of-concept for future investigations.

2.3.2 Experimental equipment and protocol

This section contains description of our equipment, test protocol, data acquisition software and data processing approaches.

Acquisition system

As it was already mentioned, a proposed acquisition device was an armband consisting of eight blocks connected by an elastic rubber band. Each block has a differential surface EMG electrode on the side opposed to the skin. It also contains an integrated IMU providing acceleration and orientation data. The armband hardware itself performs filtering and sampling of the signals (EMG at 200 Hz and IMU at 50Hz, 10bit quantisation) and may be connected wirelessly to a PC using Bluetooth Low Energy protocol.

A custom acquisition software with graphical user interface (figure 2.7) was developed to conduct the acquisition. It handled several major tasks: management of acquisition database, indication of correct armband placement and real-time signal visualisation. Back-end, or communication part of the software was developed using myo-python library [40], that provides a Python wrapper for original MYO libraries. Signals from all the channels were visualized during the acquisition, and then stored to a database.

Acquisition protocol

During tests, the MYO armband was placed either on the dominant forearm or on the lower leg, depending on the test. While on the forearm, the armband was placed

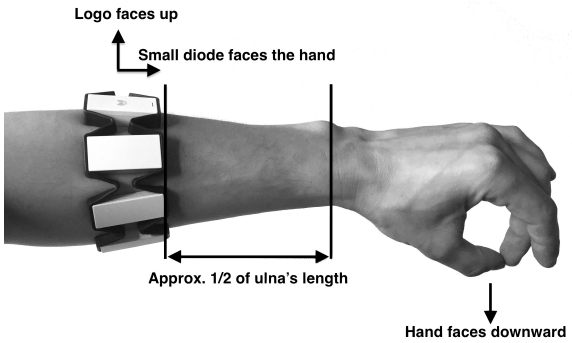


Figure 2.6: Instruction for armband positioning on the forearm, example shown for finger tapping test.

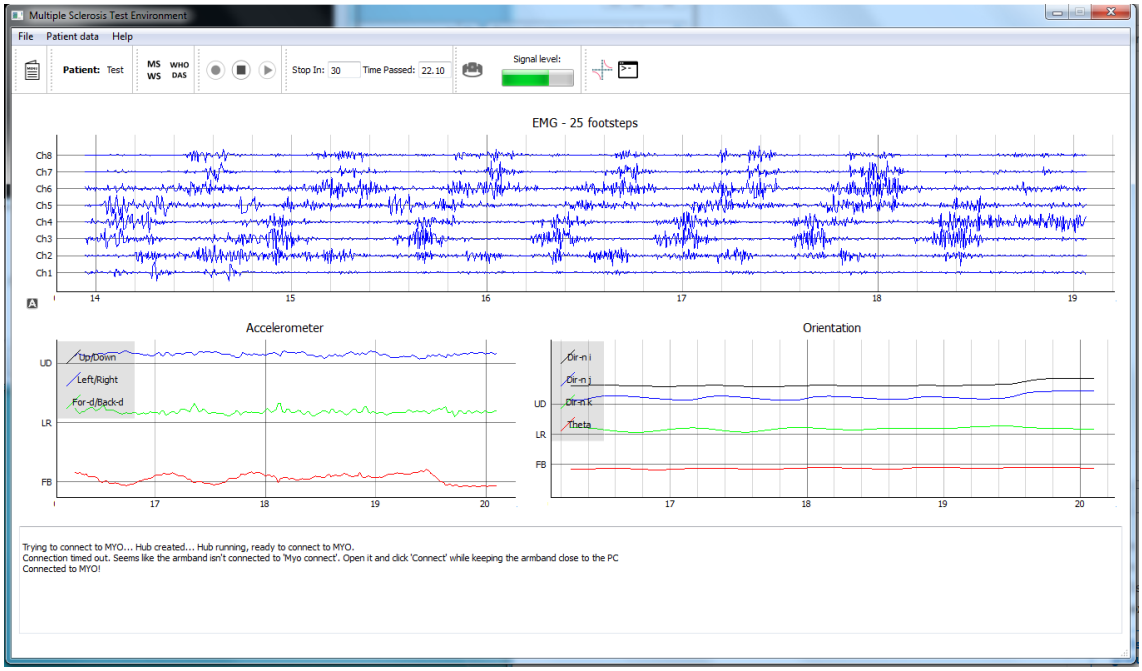


Figure 2.7: Graphical user interface developed for multiple sclerosis data acquisition.

so that its distal side was approximately in the middle of ulna bone and its logo-block was aligned with virtual line passing through middle finger (figure 2.6). On the lower leg, its distal side was set in the middle of tibia, with logo-block placed laterally to it, covering tibialis anterior muscle. In function of limb size armband was loosened or tightened using special clips to assert comfort of the subject and firm contact of EMG electrodes to the skin. Electrodes were evenly distributed around the limb.

The dataset for this publication was obtained during clinical follow-up of MS patients at Oslo University Hospital, Oslo, Norway. It consisted of 40 healthy controls and 31 MS patients, who gave their informed consent according to local guidelines. Tables 2.4 and 2.5 provide demographics and clinical evaluation statistics of the participants. MS patients were not selected to fit a specific range of disease severity

Table 2.4: Demographics of participants.

Variable, mean (SD)	**Patients N = 31**	**Healthy Controls N = 40**
Age	37.7 (7.4)	33.3 (6.8)
Gender (% of females)	70	75
Height	170.2 (7.5)	171.2 (8.8)
Weight	71.4 (16.4)	67.1 (12.4)
Right handed (%)	86.7	90

Table 2.5: Clinical evaluation of patients.

Variable, mean (SD, min-max)	**Value, patients**
Age of first symptoms	28.1 (7.7, 12 - 42)
Age of MS onset	32.5 (7.5, 19 - 49)
Disease duration	5.7 (2.6, 1.3-12.0)
EDSS	2.1 (1.1, 0.0 - 4.0)
Total number of attacks	2.3 (1, 1 - 5)

or to have similar symptoms.

Each subject performed four motor function tests: timed 25 foot walk (T25FW), finger tapping (FIT), finger-to-nose (FTN), foot tapping (FT). Subjects were instructed about each test protocol and were given time to train. Within each of the tests, at least two acquisition trials were performed.

Dataset

MS patients and healthy controls (HC) formed two classes labelled by "1" and "0" respectively. Each of the four tests was used to form a separate dataset for a separate classification problem. Within the tests, each valid trial represented a sample. Later, for each person, a classification score obtained for all trials of a selected test were averaged to obtain a final score.

2.3.3 Classification

Feature extraction

Signals from all the trials were first visually analysed and regions of interest (ROIs), containing only periodic part of the signal, were then manually extracted. This was done in order to standardise the signals in which motion and acquisition onset times were not synchronised and which in some cases contained acquisition artifacts. An automatic ROI extraction procedure would be hard to implement because of the variety of signal and artifact forms.

ROI lengths were then equalised by trimming longest signals. After, from each EMG channel, twelve temporal features were extracted [39]: number of zero crossings, waveform length (WL), slope sign change, Wilson amplitude (WA), root mean

Table 2.6: Features most represented in first principal components.

Test	Features (EMG channels)
T25FW	WL(1), WA(1), EN(8, 1, 5, 2, 4, 6)
FT	WL(4, 5), WA(4, 5), EN(2, 7, 5, 1, 8)
FIT	WL(2, 1), WA(1, 2), EN(8, 7, 5), MNF(0-20Hz)(1,2,3,5,6)
FTN	WA(3, 5, 6, 4), WL(3, 5, 6, 4), EN(1, 6), IEMG(2, 4, 5), MAV(2, 4, 5)

square, mean absolute value (MAV), integrated EMG (IEMG), signal variance and auto-regressive coefficients (4th order model).

Twelve spectral features were extracted as follows: the whole spectral band of the signal was partitioned into three sub-bands (0-20Hz, 20-50Hz, 50-100Hz), within each band mean and median frequencies (MNF and MDF) were calculated, as well as energy (EN) and dispersion of spectrum. As for IMU, only accelerometer data was taken into account, extracted features were mean and median frequencies, without band partitioning.

Dimensionality reduction

Overall number of sample descriptors was 24 for each of eight EMG channels plus two features for each of three accelerometer channels, giving an overall dimensionality of 198 for each sample. In order to reduce the dimensionality of the dataset, PCA was applied to the feature matrix of each test. Depending on the test, from six to ten principal components were kept. Features that were mostly represented in first principal components are EMG features: Wilson amplitude (WA), waveform length (WL) and energy (EN) in all three sub-bands (listed in table 2.6).

Classification model

Support vector machine (SVM) models were learned from four datasets corresponding to each of the tests, using svmlib library [28]. SVM scores for each patient were obtained during 3-fold partitioning procedure in which 1/3 of scores were predicted using other 2/3 of data as training set. Within each partition all trials of a given patient were considered as separate samples. This way, during prediction phase, patient's score was computed as average across all of his trials. Polynomial kernel was used, best values of kernel and of regularization parameter were found using grid-search over 10-fold cross-validation. The value of SVM score may be interpreted as distance between the sample and separating hyperplane, while its sign represents estimated class label.

2.3.4 Results

For each data sample (single test trial), SVM model output characterises the sample score, probability of being either healthy or having MS. For each subject, scores

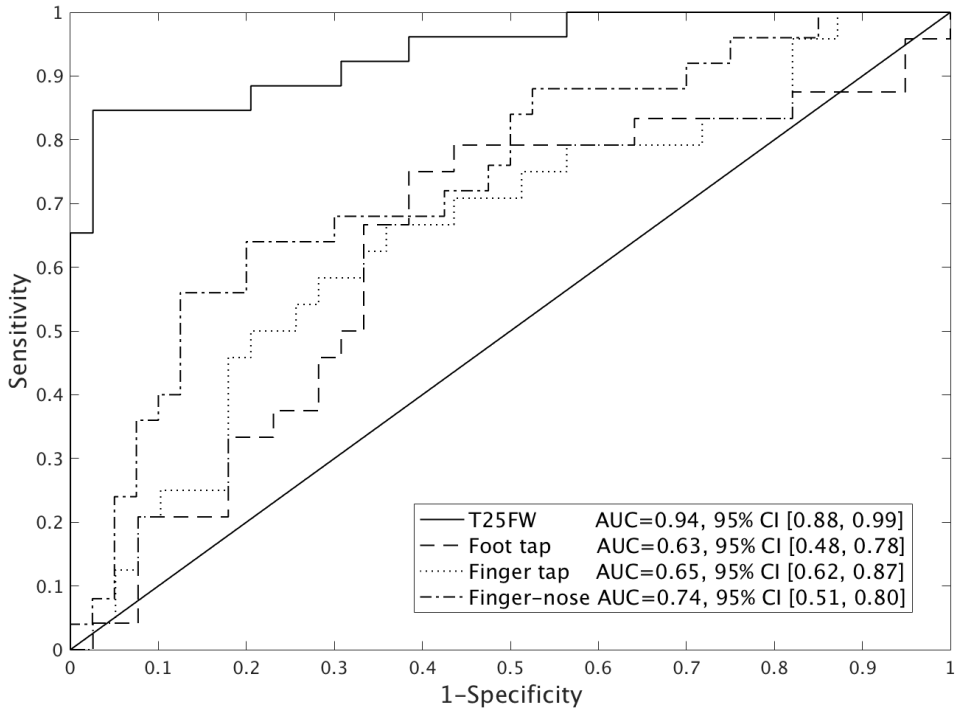


Figure 2.8: ROC curves for MS-healthy classification on four motor function tests. T25FW stands for timed 25 foot walk.

corresponding to multiple trials of the same test were averaged. Then, sigmoidal function was applied to map the average scores onto the [0,1] interval to then trace ROC curves for each of four tests (figure 2.8). Best performance obtained was for T25FW (area under ROC 95% CI [0.88, 0.99]).

The obtained scores may be interpreted differently, depending on a classification strategy. Therefore, in the situations where one single decision (for example, "no MS"/"possible MS") must be provided, the results obtained from the different tests should be combined. As an example of a possible approach, we propose a rule that consists of diagnosing a subject to have MS if 1) his most positive score is larger, by its absolute value, than his scores for the other tests; or 2) if he has at least two positive scores. Confusion matrix for that approach reveals its high true positive rate (table 2.7). The fact that overall number of MS subjects in table 2.7 sums up to 27 instead of 31 (initial number of MS participants) is due to the fact that combined classification was not performed on four patients that lacked valid trials for at least one test. However, available within-test scores of these patients were included in ROC analysis (figure 2.8).

Table 2.7: Confusion matrix for combined classification approach.

	Healthy controls	Multiple Sclerosis
Healthy controls	30	10
Multiple Sclerosis	2	25

2.3.5 Conclusion and future work

In this study we assessed the classification capabilities of MYO armband and typical EMG processing pipeline, applied to a sample of MS patients and healthy subject. Timed 25 feet walk test has shown a superior performance (in terms of the area under ROC curve) compared to other tests. Possible explanation is the fact that walking requires coordination of the whole body and thus may be affected by MS-caused abnormalities in any body part. Other tests involve independent movement of either upper or lower limb and thus may not represent the actual state of a subject if his/her MS manifests elsewhere.

Possible development of this study have to consider, first of all, the acquisition protocol. In order to compare the data acquired by two different groups, a common protocol should be established, including subject's position and trajectory of walking. Also, special care should be taken considering the intra-class variability of the EMG signals caused by different skin and body fat conditions. Necessity of EMG normalisation with respect to maximum voluntary contraction level should be investigated. No common diagnostic protocol involving MYO armband may be established until these questions are assessed.

Another mean of development will be a thorough feature selection using including ANOVA tests. Finally, as possible follow-up based on this study we are investigating the possibility of predicting MS disability based on regression of SVM scores on EDSS (Expanded Disability Status Scale) [41]. For that purpose, another dataset consisting only of MS patients with different levels of disability based on EDSS is currently being acquired.

2.4 General conclusions on the MYO armband

MYO armband by Thalmic Labs is a simple in use, robust and accessible device, which rapidly gained popularity both in academia and amateur community. A large number of various libraries and bindings was created, making the armband usable in more and more different ways. There were numerous papers on gesture recognition, prosthetic control and human interfacing based on this device, in which we have also contributed.

However, we note some of its disadvantages that are important for the research. First is an uncontrollable gain, resulting in saturation of EMG during strong contractions. Second is a small sampling frequency (200 Hz) which makes questionable the extraction of temporal and spectral features of EMG. Third, inability to access some of the important functionality without using low-level programming, while this access itself is not clearly documented. The last one is the fact that since September

2018 this device is no longer produced and no-longer supported by its developers.

However, MYO armband is still very well-suited for amateur use and gesture recognition. Additionally, it has significantly promoted the EMG technology in general, making public more familiar with it. Finally, the possible analogues coming to market after the stop of its official sales may address some of the aforementioned issues, making this technology a strong basis for further development of EMG-based methods.

Bibliography

[1] A. Atasoy, E. Kaya, E. Toptas, S. Kuchimov, E. Kaplanoglu, and M. Ozkan, "24 DOF EMG controlled hybrid actuated prosthetic hand," in *Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of The*, pp. 5059–5062, IEEE, 2016.

[2] J. G. Abreu, J. M. Teixeira, L. S. Figueiredo, and V. Teichrieb, "Evaluating sign language recognition using the myo armband," in *2016 XVIII Symposium on Virtual and Augmented Reality (SVR)*, pp. 64–70, 2016.

[3] A. R. Fonseca, *Smart control of a soft robotic hand prosthesis*. PhD thesis, Paris West University Nanterre La Défense, 2016.

[4] A. Atasoy, E. Kaya, E. Toptas, S. Kuchimov, E. Kaplanoglu, and M. Ozkan, "24 DOF EMG controlled hybrid actuated prosthetic hand," in *Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annu. Int. Conf. of the*, pp. 5059–5062, IEEE, 2016.

[5] K. Akhmadeev, E. Rampone, T. Yu, Y. Aoustin, and E. Le Carpentier, "A testing system for a real-time gesture classification using surface EMG," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 11498–11503, 2017.

[6] X. Zhang, X. Chen, W.-h. Wang, J.-h. Yang, V. Lantz, and K.-q. Wang, "Hand gesture recognition and virtual game control based on 3d accelerometer and EMG sensors," in *Proc. of the 14th Int. Conf. on Intelligent User Interfaces*, IUI '09, (New York, NY, USA), pp. 401–406, ACM, 2009.

[7] C. S. L. Tsui, P. Jia, J. Q. Gan, H. Hu, and K. Yuan, "EMG-based hands-free wheelchair control with eog attention shift detection," in *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE Int. Conf. on*, pp. 1266–1271, Dec 2007.

[8] M. DiCicco, L. Lucas, and Y. Matsuoka, "Comparison of control strategies for an EMG controlled orthotic exoskeleton for the hand," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE Int. Conf. on*, vol. 2, pp. 1622–1627 Vol.2, April 2004.

[9] A. Fougner, O. Stavdahl, P. J. Kyberd, Y. G. Losier, and P. A. Parker, "Control of Upper Limb Prostheses: Terminology and Proportional Myoelectric Control; A Review," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, pp. 663–677, Sept. 2012.

[10] M. Hakonen, H. Piitulainen, and A. Visala, "Current state of digital signal processing in myoelectric interfaces and related applications," *Biomedical Signal Processing and Control*, vol. 18, pp. 334–359, Apr. 2015.

[11] K. Xing, P. Yang, J. Huang, Y. Wang, and Q. Zhu, "A real-time EMG pattern recognition method for virtual myoelectric hand control," *Neurocomputing*, vol. 136, pp. 345–355, July 2014.

[12] A. J. Young, L. H. Smith, E. J. Rouse, and L. J. Hargrove, "Classification of simultaneous movements using surface EMG pattern recognition," *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 1250–1258, May 2013.

[13] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Systems with Applications*, vol. 40, pp. 4832–4840, Sept. 2013.

[14] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7420–7431, 2012.

[15] L. J. Hargrove, E. J. Scheme, K. B. Englehart, and B. S. Hudgins, "Multiple binary classifications via linear discriminant analysis for improved controllability of a powered prosthesis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, pp. 49–57, Feb 2010.

[16] M. R. Ahsan, M. I. Ibrahimy, and O. O. Khalifa, "Electromygraphy (EMG) signal based hand gesture recognition using artificial neural network (ANN)," in *Mechatronics (ICOM), 2011 4th Int. Conf. On*, pp. 1–6, May 2011.

[17] M.-F. Lucas, A. Gaufriau, S. Pascual, C. Doncarli, and D. Farina, "Multi-channel surface EMG classification using support vector machines and signal-based wavelet optimization," *Biomedical Signal Processing and Control*, vol. 3, no. 2, pp. 169–174, 2008.

[18] N. Jiang, S. Dosen, K.-R. Müller, and D. Farina, "Myoelectric control of artificial limbs - is there a need to change focus," *IEEE Signal Process. Mag*, vol. 29, no. 5, pp. 152–150, 2012.

[19] "How much does a prosthetic arm cost?," 2017. Retrieved from https://health.costhelper.com/prosthetic-arms.html on 29/07/2019.

[20] Z. Lu, X. Chen, Q. Li, X. Zhang, and P. Zhou, "A Hand Gesture Recognition Framework and Wearable Gesture-Based Interaction Prototype for Mobile Devices," *IEEE Transactions on Human-Machine Systems*, vol. 44, pp. 293–299, Apr. 2014.

[21] S. Benatti, F. Casamassima, B. Milosevic, E. Farella, P. Schonle, S. Fateh, T. Burger, Q. Huang, and L. Benini, "A Versatile Embedded Platform for EMG Acquisition and Gesture Recognition," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 9, pp. 620–630, Oct. 2015.

[22] M. Tavakoli, C. Benussi, and J. L. Lourenco, "Single channel surface EMG control of advanced prosthetic hands: A simple, low cost and efficient approach," *Expert Systems with Applications*, vol. 79, pp. 322–332, Aug. 2017.

[23] A. Boyali, N. Hashimoto, and O. Matsumoto, "Hand posture and gesture recognition using MYO armband and spectral collaborative representation based classification," in *Consumer Electronics (GCCE), 2015 IEEE 4th Global Conference on*, pp. 200–201, IEEE, 2015.

[24] M. Tomaszewski, "Myo SDK matlab MEX wrapper," 2016. Online repository.

[25] L. H. Smith, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "Determining the optimal window length for pattern recognition-based myoelectric control: Balancing the competing effects of classification error and controller delay," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, pp. 186–192, April 2011.

[26] P. Kaufmann, K. Englehart, and M. Platzner, "Fluctuating EMG signals: Investigating long-term effects of pattern matching algorithms," in *2010 Annu. Int. Conf. of the IEEE Engineering in Medicine and Biology*, pp. 6357–6360, Aug 2010.

[27] B. Hudgins, P. Parker, and R. N. Scott, "A new strategy for multifunction myoelectric control," *IEEE Transactions on Biomedical Engineering*, vol. 40, pp. 82–94, Jan 1993.

[28] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[29] M. Oskoei and Huosheng Hu, "Support Vector Machine-Based Classification Scheme for Myoelectric Control Applied to Upper Limb," *IEEE Transactions on Biomedical Engineering*, vol. 55, pp. 1956–1965, Aug. 2008.

[30] M. K. Akhmadeev, A. Houssein, M. S. Moussaoui, H. Harbo, M. E. A. Høgestøl, J. Graves, M. S. Bos-Haugen, M. P.-A. Gourraud, M. D.-A. Laplaud, and I. M. V. Tutturen, "SVM-based tool to detect patients with multiple sclerosis using a commercial EMG sensor," in *Proceedings of the IEEE 10th Sensor*

Array and Multichannel Signal Processing Workshop (SAM), 2018, Sheffield, UK, p. 4, IEEE, 2018.

[31] W. I. McDonald, A. Compston, G. Edan, D. Goodkin, H. P. Hartung, F. D. Lublin, H. F. McFarland, D. W. Paty, C. H. Polman, S. C. Reingold, M. Sandberg-Wollheim, W. Sibley, A. Thompson, S. van den Noort, B. Y. Weinshenker, and J. S. Wolinsky, "Recommended diagnostic criteria for multiple sclerosis: Guidelines from the International Panel on the diagnosis of multiple sclerosis," *Annals of Neurology*, vol. 50, pp. 121–127, July 2001.

[32] C. H. Polman, S. C. Reingold, B. Banwell, M. Clanet, J. A. Cohen, M. Filippi, K. Fujihara, E. Havrdova, M. Hutchinson, L. Kappos, F. D. Lublin, X. Montalban, P. O'Connor, M. Sandberg-Wollheim, A. J. Thompson, E. Waubant, B. Weinshenker, and J. S. Wolinsky, "Diagnostic criteria for multiple sclerosis: 2010 Revisions to the McDonald criteria," *Annals of Neurology*, vol. 69, pp. 292–302, Feb. 2011.

[33] M. Kaufman, D. Moyer, and J. Norton, "The significant change for the Timed 25-Foot Walk in the Multiple Sclerosis Functional Composite," *Multiple Sclerosis Journal*, vol. 6, pp. 286–290, Aug. 2000.

[34] S. H. Alusi, J. Worthington, S. Glickman, and P. G. Bain, "A study of tremor in multiple sclerosis," *Brain*, vol. 124, pp. 720–730, Apr. 2001.

[35] I. Jonkers, G. Nuyens, J. Seghers, M. Nuttin, and A. Spaepen, "Muscular effort in multiple sclerosis patients during powered wheelchair manoeuvres," *Clinical Biomechanics (Bristol, Avon)*, vol. 19, pp. 929–938, Nov. 2004.

[36] S.-S. Ayache, T. Al-ani, W.-H. Farhat, H.-G. Zouari, A. Créange, and J.-P. Lefaucheur, "Analysis of tremor in multiple sclerosis using Hilbert-Huang Transform," *Neurophysiologie Clinique = Clinical Neurophysiology*, vol. 45, pp. 475–484, Dec. 2015.

[37] K. J. Kelleher, W. Spence, S. Solomonidis, and D. Apatsidis, "The characterisation of gait patterns of people with multiple sclerosis," *Disability and Rehabilitation*, vol. 32, pp. 1242–1250, Jan. 2010.

[38] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Systems with Applications*, vol. 40, pp. 4832–4840, Sept. 2013.

[39] M. Hakonen, H. Piitulainen, and A. Visala, "Current state of digital signal processing in myoelectric interfaces and related applications," *Biomedical Signal Processing and Control*, vol. 18, pp. 334–359, Apr. 2015.

[40] N. Rosenstein, "myo-python: Python bindings for the myo SDK." Online repository.

[41] J. F. Kurtzke, "Rating neurologic impairment in multiple sclerosis: An expanded disability status scale (EDSS)," *Neurology*, vol. 33, no. 11, pp. 1444–1444, 1983.

Chapter 3

Models for intent estimation from EMG decomposition

3.1 Introduction and main definitions

3.1.1 Organization of motor control

Each motor function of the human body is carried out by a complex joint action of skeletal muscles. Degrees and velocities of their contraction fully define the outcome of the movement: its amplitude, speed, and trajectory. The contraction of each skeletal muscle is regulated by the activity of spinal α motor neurons (MN) that innervate it.

MNs that innervate the muscle form a functional group that is called *motor neuron pool* (MNP). They elicit a contraction of the muscle by generating action potential trains, also called *firings* or *spike trains*. Generally, the more action potentials the muscle receives per a unit of time, the more intense the contraction is. Therefore, there exist two mechanisms of contraction force regulation: recruitment/de-recruitment of MNs and firing frequency variation.

The order in which the recruitment or de-recruitment are performed is structured so that each MN has a specific level of contraction above which it fires, called *recruitment threshold* (RT). According to Henemann's principle [1], MNs with larger physical sizes and larger contributions to the contraction force tend to have higher RTs.

Once a MN is recruited, its contribution to the total contraction force is regulated by the frequency of its firings, or *firing rate*. Experimental studies show that firing rates of MNs of the same pool exhibit a significant covariation during the changes of contraction force [2]. That permits to suppose that they receive a significant proportion of common excitatory input, usually referred to as *common drive*.

Common drive largely defines the behavior of MNs of the entire pool and therefore defines the contraction force. An increase in the common drive leads to the recruitment of new MNs and an increase in firing rates in the already recruited ones. This permits to state that the contraction force is an increasing, generally non-linear, function of the common drive.

In fact, this principle can be applied to other mechanical manifestations of the movement, that are increasing functions of the muscle contraction force, such as joint angle, joint torque, or some more complex motor functions. In the following, we are going to refer to all of them as the mechanical effect, or simply the *effect*, and denote it as e.

3.1.2 Model that relates the effect to the MN firing behavior

As we have stated earlier, the effect and the firing behavior of MNs are defined by the same source, which is the common drive. Establishing a relation between them is one of the major subjects in motor control studies. Usually, such relation is established between the contraction force and firing rate and is modeled as a linear function, starting at the MN's recruitment threshold [3, 4] and reaching saturation at higher forces [5]:

$$\varphi_i(e) = \begin{cases} \min\left(\varphi_i^{\max},\ g_i(e - r_i) + \varphi_i^{\min}\right), & e > r_i \\ 0, & e \leq r_i \end{cases} \tag{3.1}$$

where φ_i is firing rate of i-th MN, e is, in this case, the force of contraction; $\varphi_i^{\min}$ and φ_i^{max} are the minimum and maximum firing rates; r_i is its recruitment threshold and g_i is the slope of linear relation. Other approaches may include non-linear models, such as one presented in [6].

In our work, we will adopt the linear modelling approach. We will also suppose that it can be applied not only to the contraction force but to other types of the effect. To further adapt it to our application, we will not take into consideration the saturation of effect-rate curve, since it appears only at high contraction levels, unreachable for currently available decomposition algorithms. Additionally, we will break down this model into two parts by introducing a binary *activation* variable a representing the MN's recruitment state:

$$a_i(e) = \begin{cases} 1 & \text{if } e > r_i \\ 0 & \text{if } e \leq r_i \end{cases} \qquad \varphi_i(e) = a_i\left(g_i e + b_i\right) \tag{3.2}$$

The introduction of a is redundant since it exactly copies the on-off relationship between the firing rate and the effect. However, we will show later how a can be derived from the decomposition data, and that the effect inference can be based solely on this variable. While building an effect estimator based on the instantaneous firing rates, we will show how to eliminate this variable from the inference.

3.1.3 Definition of the intended effect estimation problem and main assumptions

Experimental studies show that common drive is influenced by proprioceptive feedback from muscle spindles and Golgi tendon organs [7, 8] as well as by concurrent antagonist and agonist muscles contractions [9, 10]. That is, the common drive

to a MNP is not only a function of supraspinal command, representing the intent but also of multiple spinal level interactions with other pools and with the muscle. However, we will suppose that the common drive to a MNP is still predominantly defined by the supraspinal command and that elicited spike trains of the MNs are functions of the intent.

After amputation, the biomechanical end-effector, such as hand, is missing, while the neural input to the residual muscles that actuate it, is still present. Thus, we seek to establish a system that decomposes the spiking activity of MNs from the EMG of residual muscles, estimates the intended effect and hands it to the mechanical part of the prosthesis. Together, these elements should provide the best possible approximation of the original end-effector's movement. In this work, we concentrate on the problem of estimation of the intent, which can be then transferred to the motors of the mechanical part in order to realize the intended movement.

Given a formulation of an intent estimation model, one should identify its parameters when applying it to a new user or motor function. To do it, training data are required, which consist of the intended effect measures, taken synchronously with the decomposed spike trains. However, the intended effect itself is not observable, since supraspinal command cannot be (or is very impractical to be) accessed. Therefore we will accept, as it is generally done in other prosthetic control strategies, that observed effect *is equivalent* to the intended one, and therefore can be used to identify an intent estimation model.

Observation of the effect is not possible in the amputees, and therefore the problem of training data collection arises. In this case, bilateral training approach is possible, in which the subject tries to symmetrically move both the phantom and the intact limbs (mirrored bilateral training [11]). The effect is observed from the last, while the residual muscles of the missing limb provide the EMG data. Another approach consists in the subject following a virtual effect trajectory with his or her phantom limb (prosthesis-guided training [12]).

3.1.4 General approach to the decomposition-based effect estimation problem

The classic objective of myoelectric control is to establish a model that permits to infer the intended mechanical effect from the parameters of the EMG signal. Considering this and the previous section, the general strategy of decomposition-based control can be the following. Decomposition identifies the firing instants of MNs, which permits to tell which MNs are currently active and to estimate their instantaneous firing rates. This information can be passed to the inverse of a previously introduced effect-rate model in order to infer the effect.

As follows from the effect-rate model (3.2), the effect can theoretically be fully determined by the firing rate of one single MN, once it is recruited. In an idealistic scenario, where the MNs receive only noise-free common drive, and where firing rates can be perfectly estimated from decomposition, all the estimates of effect made from active MNs will be in consensus (see Figure 3.1).

However, since alongside with the common drive, MNs receive noisy individual

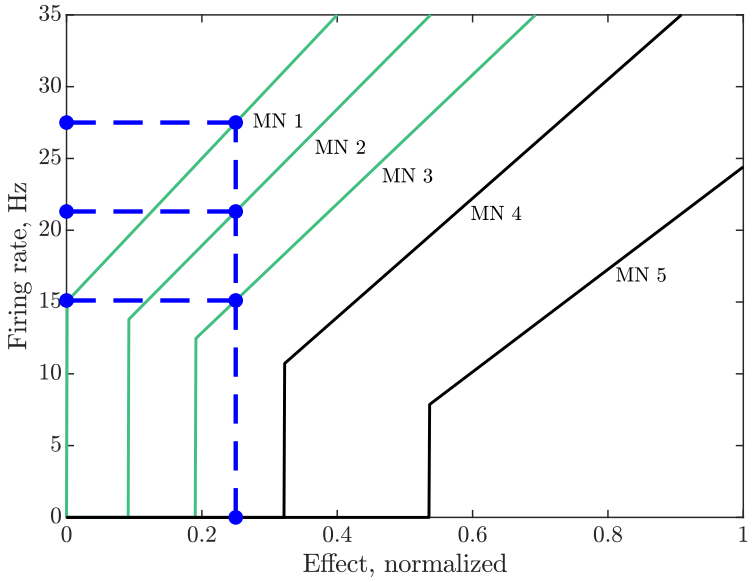


Figure 3.1: Estimates of the effect given by individual effect-rate models (3.2) and firing rates of each decomposed MN, coincide in an idealistic scenario where the estimated firing rates contain no noise or perturbations

neural inputs, observed IPIs are random. Additionally, neural command to the muscle can change very fast, so that the current contraction can be defined by a very short time span of the most recent spike trains. This limits the number of IPIs available for the averaging when estimating the firing rates. As an example, the typical window of 250 ms, chosen for the decision taking in prosthetic control, may contain a very limited amount of IPIs (approximately from one to six). Moreover, estimation of firing rate is not possible until at least one IPI (i.e., two spikes) is available, which may introduce a significant delay to the estimation at low firing rates.

Due to the joint impact of these factors, the estimates of firing rate can contain a significant amount of noise, which leads to the inability of producing a viable estimate of the effect from one the spike train of one single MN. Moreover, estimates from other MNs will not be in consensus anymore. Thus, we look for a way to effectively merge them into one, providing a more precise result.

To achieve this, in the following sections of this chapter, we will establish three probabilistic models that link the decomposed spike trains with the effect. Specifically, we will first establish a model that infers the effect solely from the recruitment of MNs (Section 3.2). Secondly, we will present an approach to use the effect-rate model in order to obtain a finer estimate of the effect (Section 3.3.1). Finally, we will merge these two models, combining the advantages of both (Section 3.3.6).

3.2 Inference of the intended effect from the motor neurons recruitment

3.2.1 Probabilistic model of MN recruitment

The value of effect at which the MN starts to fire varies across trials of the same contraction profile [13]. To account for this variability, we will proceed by modelling the recruitment threshold (RT) as a random variable instead of a fixed one.

In [13], the distribution of the RTs was reported as "quasi-normal", thus, to model the RT distribution, we consider the family of unimodal location-scale probability density functions (PDFs). Let us denote this PDF and the corresponding cumulative distribution function (CDF) as $w(r_i, \omega_i)$ and $W(r_i, \omega_i)$, where ω_i is vector containing the location and scale parameters. A concrete choice of the RT PDF is not necessary for establishing the probabilistic model and will be provided later in this section.

Let us denote the RT of i-th MN as R_i. Here and further the uppercase letters (E, R_i) will denote the random variables, while their lowercase versions (e, r_i) will denote values that these variables can take. In cases of distributions, uppercase and lowercase letters will denote, respectively, CDFs and PDFs.

Let us denote the fact that i-th MN is recruited, or active, by a random binary variable A_i. This variable is a function of RT R_i and of the intended effect E:

$$A_i = \begin{cases} 1 & \text{if } R_i \leq E \\ 0 & \text{if } R_i > E \end{cases}$$

Relationship between R_i, A_i and E can be represented by a graph illustrated in Figure 3.2. Probabilistic formulation $\Pr(A_i|E, R_i)$ of this relationship can be represented as follows:

$$\begin{aligned} \Pr(A_i = 1|E, R_i) &= \begin{cases} 1 & \text{if } R_i \leq E \\ 0 & \text{if } R_i > E \end{cases} \\ \Pr(A_i = 0|E, R_i) &= \begin{cases} 0 & \text{if } R_i \leq E \\ 1 & \text{if } R_i > E \end{cases} \end{aligned} \tag{3.3}$$

The intended effect E is the target variable that we want to estimate in order to control a prosthesis. Activation A_i, being an output of the EMG decomposition algorithm, can be considered the observation from which we will infer the target variable. Relationship between the two is established by recruitment model (3.3) with its parameter R_i distributed according to PDF $w(r_i, \omega_i)$.

More specifically, we are interested in the conditional distribution $\Pr(A_i|E, \Omega_i)$. It can be calculated using RT CDF $W(r_i, \omega_i)$ and (3.3) using total probability rule, as well as the facts that $\Pr(R_i \leq E|\Omega_i) = W(E, \omega_i)$ and $\Pr(R_i > E|\Omega_i) = 1 - W(E, \Omega_i)$ (see Appendix A for full derivation):

$$\Pr(A_i = a_i|E, \Omega_i) = \begin{cases} W(E, \Omega_i) & \text{if } a_i = 1 \\ 1 - W(E, \Omega_i) & \text{if } a_i = 0 \end{cases} \tag{3.4}$$

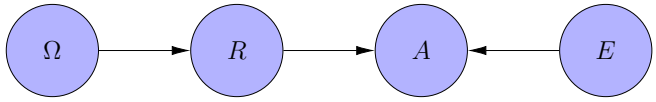


Figure 3.2: Bayesian graph representing the probabilistic model of recruitment of a MN. Parameter Ω define the distribution of RT R. In its turn, RT, together with the intended effect E, define the activation of the MN A.

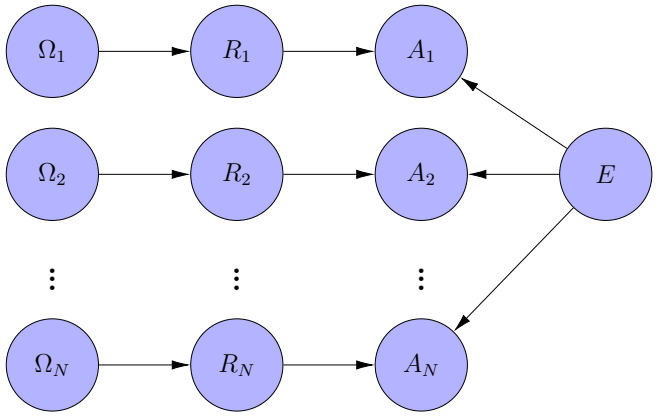


Figure 3.3: Bayesian graph representing the probabilistic model of recruitment of multiple MNs. Probabilities of activation in each MN are independent, given the effect E.

3.2.2 Recruitment model for multiple MNs

Model 3.4 describes the probability that MN is recruited, given the current effect value and parameters of its RT distribution. To establish a joint probabilistic model for multiple MNs, we will make a number of assumptions about the corresponding distributions. We illustrate these assumptions using the Bayesian graph provided in Figure 3.3.

That is, we assume that the distributions $\Pr(A_i|E,\Omega_i)$ for all the MNs $i \in \{1\!:\!N\}$ are conditionally independent given E. Therefore, joint distribution $\Pr\left(A|E,\Omega\right)$ is a product of the individual ones. By defining vectors of parameters $A = \{A_1, A_2 \ldots, A_N\}$ and $\Omega = \{\Omega_1, \Omega_2 \ldots, \Omega_N\}$, we have:

$$\Pr\left(A = \{a_1, a_2, \ldots, a_N\}|E,\Omega\right) = \prod_{i=1}^{N} \Pr(A_i = a_i|E,\Omega_i) \tag{3.5}$$

which is a joint function likelihood function for RT distribution parameters Ω and effect E.

Using this formulation, we can solve the following problems:

- *Model identification*, or estimation of parameters Ω of the distribution (3.5). This can be achieved by maximization of likelihood $P(A|E,\Omega)$ over a training dataset $\mathcal{D} = \{(a,e)^{(1)}, (a,e)^{(2)}, \ldots, (a,e)^{(D)}\}$. It is common to assume that the training samples in $\mathcal{D}$ are independently and identically distributed, given

parameters Ω. The joint likelihood over the data set is a product of likelihoods for each sample:

$$L(\mathcal{D}, \omega_1, \omega_2, \ldots, \omega_N) = \prod_{i=1}^{N} \prod_{d=1}^{D} \Pr(A_i = a_i^{(d)} | E = e^{(d)},\ \Omega_i = \omega_i) \tag{3.6}$$

Thus, the negative log-likelihood cost function:

$$C(\mathcal{D}, \omega_1, \omega_2, \ldots, \omega_N) = -\sum_{i=1}^{N} \sum_{d=1}^{D} \log \Pr(A_i = a_i^{(d)} | E = e^{(d)},\ \Omega_i = \omega_i) \tag{3.7}$$

can be optimized separately for each motor neuron i, to estimate its RT distribution parameters, by minimizing the corresponding negative log-likelihood:

$$C_i^T(\mathcal{D}, \omega_i) = -\sum_{d=1}^{D} \log \Pr(A_i = a_i^{(d)} | E = e^{(d)},\ \Omega_i = \omega_i) \tag{3.8}$$

$$\hat{\omega}_i = \underset{\omega_i}{\operatorname{argmin}}\, C_i^T(\mathcal{D}, \omega_i)$$

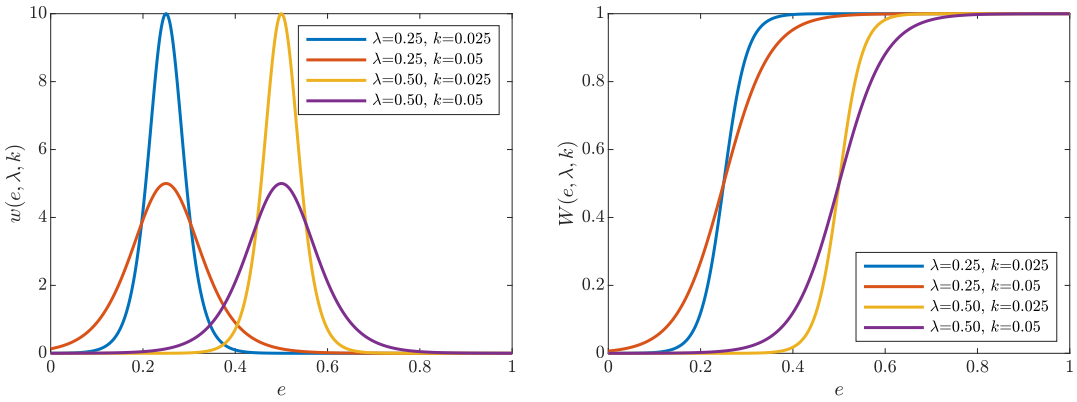
- *Effect inference*, or estimation of the effect by minimization of joint negative log-likelihood (3.5) given $A = a$ and the identified $\Omega = \hat{\omega}$:

$$C^I(a, e, \hat{\omega}_1, \hat{\omega}_2, \ldots, \hat{\omega}_N) = -\sum_{i=1}^{N} \log \Pr(A_i = a_i | E = e,\ \Omega_i = \hat{\omega}_i) \tag{3.9}$$

$$\hat{e} = \underset{\omega_i}{\operatorname{argmin}}\, C^I(a, e, \hat{\omega}_1, \hat{\omega}_2, \ldots, \hat{\omega}_N)$$

Training set $\mathcal{D}$ includes values of the effect and of the MN activation acquired synchronously during the training phase, using either mirror bilateral or prosthesis-guided approaches. The activation is extracted from the decomposition of the iEMG that accompanies the effect. In the inference phase, we use the model parameters estimated during the training phase to infer the intended effect from the available online decomposition data.

Both tasks of model identification and of effect inference can be solved by minimization of their negative log-likelihood cost functions (3.8) and (3.9). An approach to that will be presented in the following sections. Before that, we will specify the type of PDF that we use to model the distribution of the RT. Let us address this question in the next section.



(a) Logistic PDF (b) Logistic CDF

Figure 3.4: Examples of logistic probability and cumulative distribution functions. As can be seen, parameter k sets the position of the mode, while λ defines the shape of the distribution.

3.2.3 Choice of PDF for RT distribution

Until now, we have not specified the actual type of RT PDF, only requiring it to be unimodal of location-scale family. In this section, we will narrow down the class of acceptable PDFs and propose *logistic distribution* as one of the classic choices.

Equation (3.4) defines the conditional probability of a MN to be active using the CDF of RT $W(r_i, \omega_i)$. Then, logarithm of this conditional law appears in the cost functions (3.8) and (3.9). In order to effectively minimize them, it is useful to have CDF of RT provided in closed form. That excludes number of distributions, such as normal, Rayleigh, Chi-squared, etc.

Let us consider logistic law as the distribution of RT. It meets the aforementioned requirements of having location and scale parameters, as well as closed-form CDF. Also, it is widely studied in statistics and machine learning, having a number of proven useful mathematical properties, such as convexity of the corresponding ML cost function. Let us provide the corresponding CDF and PDF (also see Figure 3.4):

$$W(r_i, \lambda_i, k_i) = \frac{1}{1 + \exp\left[-\frac{r_i - \lambda_i}{k_i}\right]} \tag{3.10}$$

$$w(r_i, \lambda_i, k_i) = \frac{1}{k_i} W\left(r_i, \lambda_i, k_i\right)\left(1 - W(r_i, \lambda_i, k_i)\right) \tag{3.11}$$

where λ_i and k_i are, respectively, location and scale parameters of logistic distribution.

As a disadvantage of this choice, we note the fact that its support is $[-\infty, \infty]$, while the effect is non-negative and is bounded by a maximal value. However, as we will see from the following sections, this fact does not prevent the model from fitting to the data and providing correct inference.

3.2.4 Cost function for ML-estimation of RT distribution parameters

RT distribution parameters $\omega_i = \{\lambda_i, k_i\}$ of i-th MN can be calculated using optimization of the cost function presented in (3.8). Using (3.4), one can formulate C^T the following way:

$$C_i^T = -\sum_{d=1}^{D} a_i^{(d)} \cdot \log\left(W(e^{(d)}, \omega_i)\right) + \left(a_i^{(d)} - 1\right) \cdot \log\left(1 - W(e^{(d)}, \omega_i)\right)$$

Using Logistic CDF formula (3.10), we can rewrite it:

$$C_i^T = \sum_{d=1}^{D} a_i^{(d)} \log\left(1 + \exp\left[-\frac{e^{(d)} - \lambda_i}{k_i}\right]\right) + \left(a_i^{(d)} - 1\right) \log\left(1 + \exp\left[\frac{e^{(d)} - \lambda_i}{k_i}\right]\right) \tag{3.12}$$

Let us demonstrate the convexity of this cost function with respect to parameters λ, k. Exponent $\pm(e - \lambda)/k$, using $\tilde{\lambda} = \lambda/k, \tilde{k} = 1/k$, can be rewritten as $\pm(\tilde{k}e - \tilde{\lambda})$. The last is an affine function of $\tilde{\lambda}, \tilde{k}$ and is therefore convex. Function $f(x) = \log\left(1 + \exp(x)\right)$ is convex due to positivity of its Hessian $f''(x) = \exp(x)\left(1 + \exp(x)\right)^{-2}$. It is also an increasing function, and thus, by composition rule, $f\left(\pm(\tilde{k}e - \tilde{\lambda})\right)$ is convex with respect to $\tilde{\lambda}, \tilde{k}$. Thus finally, C_i^T is a non-negative linear mixture of convex functions and therefore is also convex. Once optimization procedure is accomplished, we get back to the original parameters using $k = 1/\tilde{k}, \lambda = \tilde{\lambda}/k$.

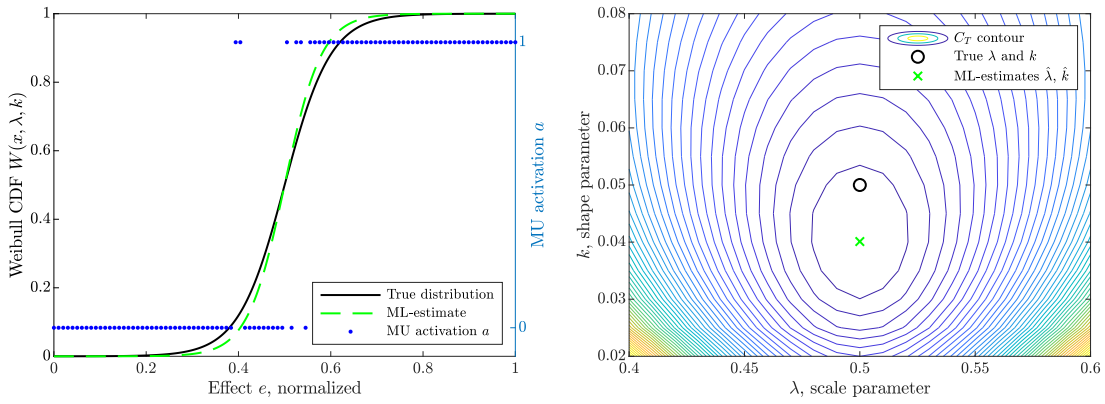
This cost function can be minimized using a standard unconstrained non-linear optimization. That is, examples presented here and later in this chapter were obtained using Matlab's *fminsearch* function. Figure 3.5 demonstrates the effect estimation for a data set simulated by the direct use of (3.4) and (3.10) with parameters λ=0.5 and k=0.05. Blue dots (see Figure 3.5(a)) designate the activation a_i observed at different values of the effect during several trials of contractions. For this example, these values were drawn from logistic distribution

We note here that in the special case of separation of samples (i.e., when are is such λ_i, k_i for which the rule $W(e_i^{(d)}, \lambda_i, k_i) \geq 1/2$ correctly identifies all samples with $a_i^{(d)} = 1$) estimation of parameter k is not possible [14]. Parameter λ_i, however, can still be estimated, and k_i can be chosen as $F_k\lambda_i$, where $F_k < 1$ is a constant proportionality coefficient.

3.2.5 Cost function for ML-estimation of effect from available MU activation data

Similarly to the training, inference cost function C^I (3.9) can be formulated the following way, using logistic CDF (3.10):

$$C^I = -\sum_{i=1}^{N} a_i \log\left(W(e, \hat{\omega}_i)\right) + (1 - a_i)\log(1 - W(e, \hat{\omega}_i))$$



(a) (b)

Figure 3.5: Example of ML-estimation of RT distribution parameters on a simulated data set containing effect samples close to the RT. (a) - Black solid is the true CDFs of RT for λ=0.5, k = 0.05; blue dots (rigth y-axes) - observed activation a of the MN from several contraction trials; green dashed line - CDF estimated from the observed activation. (b) - Contour plot of the corresponding C^T cost function.

$$C^I = \sum_{i=1}^{N} a_i \log\left(1 + \exp\left[-\frac{e - \hat{\lambda}_i}{\hat{k}_i}\right]\right) + (1 - a_i)\log\left(1 + \exp\left[\frac{e - \hat{\lambda}_i}{\hat{k}_i}\right]\right) \quad (3.13)$$

Convexity of cost function C^I with respect to e can be proved by composition rule, similarly to C^T. Since the normalized effect is constrained between zero and one, the minimum of C^I exists. Intuitively, the minimum is not on either of the borders when both active and inactive MNs are included to the inference, that is, when $0 < \sum_{i=1}^{N} a_i < N$. In cases when $\sum_{i=1}^{N} a_i = 0$ or $\sum_{i=1}^{N} a_i = N$, the estimated effect is equal to, respectively, zero or one.

As well as C^T, it can be optimized using unconstrained non-linear optimization. For illustration, we simulate a set of three MNs with specified values of λ=[0.35, 0.45, 0.55] and k=[0.01, 0.01, 0.01]. We then set the following vector of activation a=[1, 0, 0] so that the first MN is active, while the rest are inactive. Left solid dot curve in Figure 3.6 designates the position of the cost function's minimum for activation vector a=[1, 0, 0].

Next, we activate the 2-nd MN, which gives us a=[1, 1, 0]. One can see how the activation of the 2-nd MN affects the position of the minimum (Figure 3.6). In fact, each of the summands of C^I provides a soft bound for optimal value of the effect. The orientation of this bound depends on MN's activation a_i; as an example, on figure 3.6 solid and dashed (not dash-dotted) green lines correspond to the 2-nd MN's summand in C^I which imposes either a left (a_2=1) or right (a_2=0) bound to the optimal effect.

Finally, we note that $\hat{e} = f(a, \omega)$ and since a is a binary vector, there is only 2^N possible values of $\hat{e}$ for a given ω. These values can be pre-calculated for all possible combinations of a, and therefore the optimization of C^I can be replaced by very fast check in a look-up table.

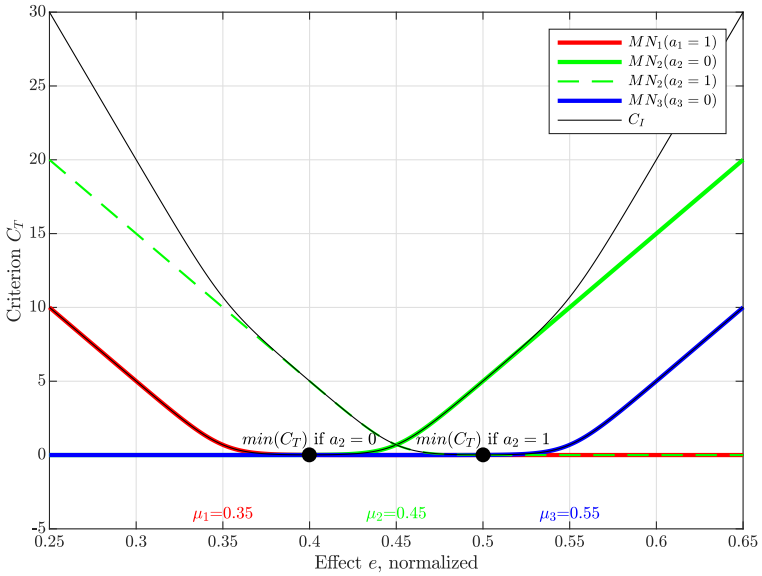


Figure 3.6: Cost function for different effect values as function of the set of active MNs. Lines "MN_1", "MN_2" and "MN_3" designate the values of the summands in C^I formula (3.13), while black line is the total sum. First black marker ($a_2 = 0$) indicates the position of the minimum for activation a=[1, 0, 0], while the second one ($a_2 = 1$) is for a=[1, 1, 0].

3.2.6 Example on simulation data

To study the characteristics of the proposed estimator, we have simulated a set of contractions using the model presented in Chapter 5 and set of parameters presented in Chapter 6. Data set included three ramp profiles with different slopes: 2.5%/s and 10%/s for training and 5%/s for testing. All contractions reached 25% MVC.

Training set was used to estimate parameters λ and k for the corresponding MNs, using training cost function (3.8). It contained spike trains of, in total, N=11 MUs. The estimate was capped to the maximal observed contraction level e_M, so that fully activated vector a, such that $\sum_n a_n = N$, gives $\hat{e} = e_M$. Another way to bound the estimate is to add a virtual $(N + 1)$-th MU with $\lambda_{N+1} = e_M$ and $k_{N+1} \ll k_N$ that is always inactive.

Testing set contained a single ramp contraction reaching 15% MVC (see Figure 3.7). Estimates of parameters λ and k were used to infer the effect by optimization of inference cost function (3.9). As one can see in Figure 3.7, the estimate remains constant in time intervals where no recruitment occurs. This behavior is expected considering the analysis provided in previous section (3.2.5).

Figure 3.8 shows the progression of the C^I cost function along the contraction ($-\log C^I$, for better illustration). One can see that cost function is flat around its minimum, similarly to Figure 3.6. This means that the position of the minimum can be easily adjusted by the inclusion of an additional term to the inference cost function.

This can be realized in order to smooth the estimate between recruitment mo-

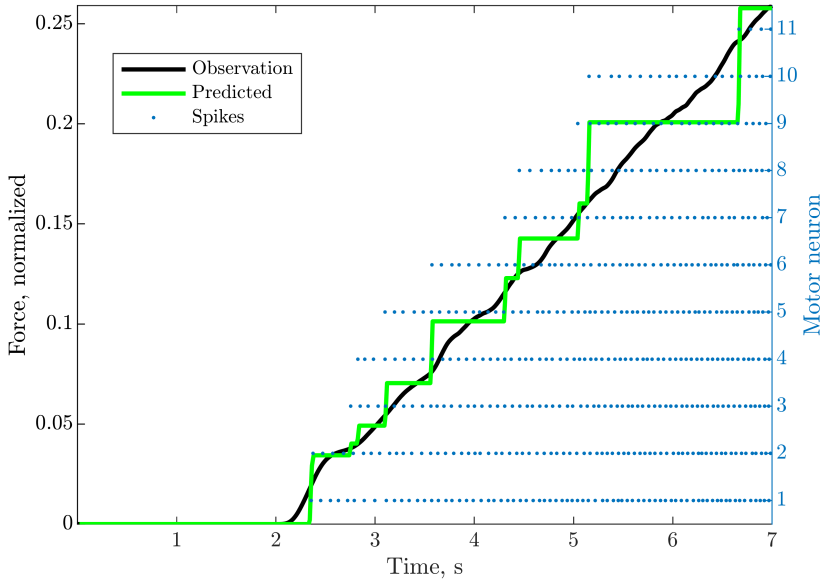


Figure 3.7: Simulated contraction force (solid black) and its estimate (solid greed) obtained by minimization of RT-based inference cost function (3.13). Blue dots designate the spike trains of MNs provided by decomposition.

ments. Such an additional term may, for example, be dependent on the instantaneous firing rate of active MNs. Or, it can be any other continuous function of the effect, such as rectified EMG amplitude. Conversely, summands of the inference cost function C^I can be considered as penalties to the effect values that are not adequate to the current activation vector.

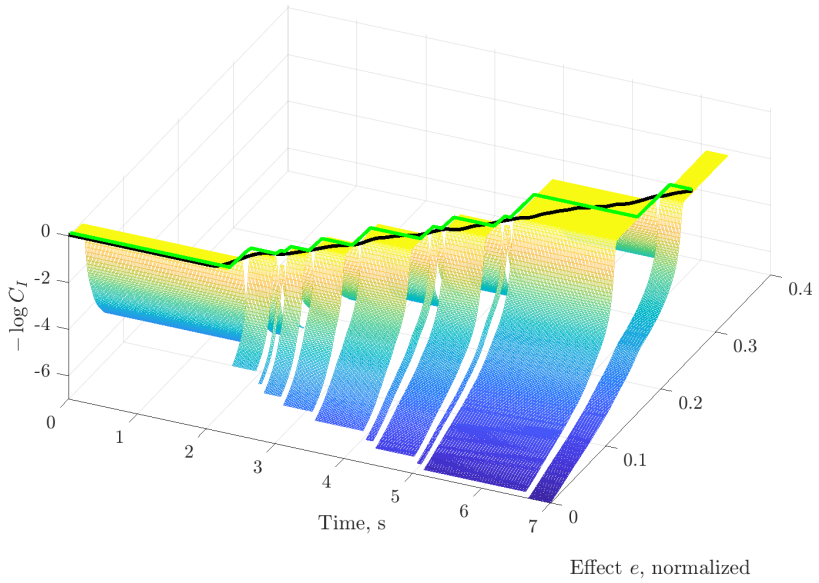


Figure 3.8: Values of cost function for RT-based inference during the contraction in Figure 3.7. Negative logarithm of cost function ($-\log(C^I)$) is shown for better illustration

.

3.3 Effect inference from firings rates of motor neurons

The estimation model proposed in the previous section utilizes the recruitment phenomena to infer the intended effect. As the provided examples show, the estimate is limited to a set of discrete values. This may not be a problem if the number of decomposable MUs is large. In that case, one may consider the gaps between the discrete estimates to be small enough for a comfortable proportional control. However, existing real-time decomposition approaches do not provide the decomposition of concurrent activity of more than 10 MUs [15]. Additionally, recruitment thresholds of observed MUs may be distributed non-uniformly, creating large gaps between consecutive discrete estimates.

Contraction force modulation is achieved by means of not only recruitment of new MUs, but also by firing rates of already recruited ones, which, until this point, were not taken into account by the estimation model. In fact, in the interval between two consecutive recruitments, the increase of excitation should result in an increase of the firing rates. This could provide additional information needed to fill the gaps between discrete estimates.

In the remainder of this chapter, we will propose a probabilistic model that incorporates the effect-rate curves to provide a continuous estimate of the intended effect. Then, we will join it with the RT distribution model in order to regularize the estimate.

3.3.1 Hidden Markov model of a spike train

In this section we partly follow the modelling approach proposed in [16, 17]. That is, we introduce a random variable Δ_i denoting an inter-pulse interval of i-th MN. Then, a sparse sequence of zeros and ones $U_i[n]$ that denotes the spike train of i-th MN in discrete time. Next, we define the *sojourn time* $T[n]$ representing the discrete amount of time passed since the last pulse of i-th MN. That is, $U[n] = \delta(T[n])$, where $\delta(\cdot)$ is Kronecker delta, and:

$$T[n+1] = \begin{cases} T[n]+1 & \text{if } U[n]=0 \\ 0 & \text{if } U[n]=1 \end{cases} \tag{3.14}$$

We will assume that inter-pulse intervals (IPIs) are independently and identically distributed following some probability density function with parameters Θ, E, functions of the intended effect E. Concrete choice of this PDF will be covered in next section. As it is shown in [17], sequence $T_i[n]$ is Markovian in this case. The corresponding transition law can be written with the help of IPI distribution law's hazard rate function $h\left(T_i[n-1], \Theta_i, E\right)$ (see definition and details in Section 3.3.2):

$$\begin{aligned} \Pr(T_i[n] = t_i[n] \mid \Theta_i, E, A_i[n]=1) &= \begin{cases} h\left(T_i[n-1], \Theta_i, E\right) & \text{if } t_i[n]=0 \\ 1-h\left(T_i[n-1], \Theta_i, E\right) & \text{if } t_i[n]=T_i[n-1]+1 \end{cases} \\ \Pr(T_i[n] = t_i[n] \mid \Theta, E_i, A_i[n]=0) &= \begin{cases} 0 & \text{if } t_i[n]=0 \\ 1 & \text{if } t_i[n]=T_i[n-1]+1 \end{cases} \end{aligned} \tag{3.15}$$

where $h(\cdot)$ is the hazard rate function $h\left(t, \Theta_i, E\right) = \Pr(\Delta_i = t \mid \Delta_i \geq t, \Theta_i, E)$ and Θ_i are the parameters of IPI distribution. Activation $A_i[n]$ has the same meaning as in previous section and can be approximated from decomposition as:

$$A_i[n] = \begin{cases} 1 & \text{if } T_i[n] < T_{\text{lim}} \\ 0 & \text{otherwise} \end{cases} \tag{3.16}$$

where T_{lim} can be chosen accordingly to the minimal observed firing rate among the MNs. Typical value can be T_{lim}=250 ms. Relation between $U[n]$, $T[n]$ and $A[n]$ is illustrated in Figure 3.9. Alternatively, activation variable can be eliminated from the inference using total probability formula and conditional law (3.4). This question will be addressed later in Section 3.3.6.

Relation between the intended effect $E[n]$, IPI distribution parameters $\Theta[n]$, activation $A[n]$ and sojourn time process $T[n]$, can be described by a hidden Markov model, illustrated using Bayesian graph in Figure 3.10. Following the modelling approach proposed in [17], we will, at first, suppose that effect is constant in time and its transition law is:

$$E[n] = E[n-1] \tag{3.17}$$

However, effect is essentially a time-varying value, thus an approach to its tracking will be presented in the sequel.

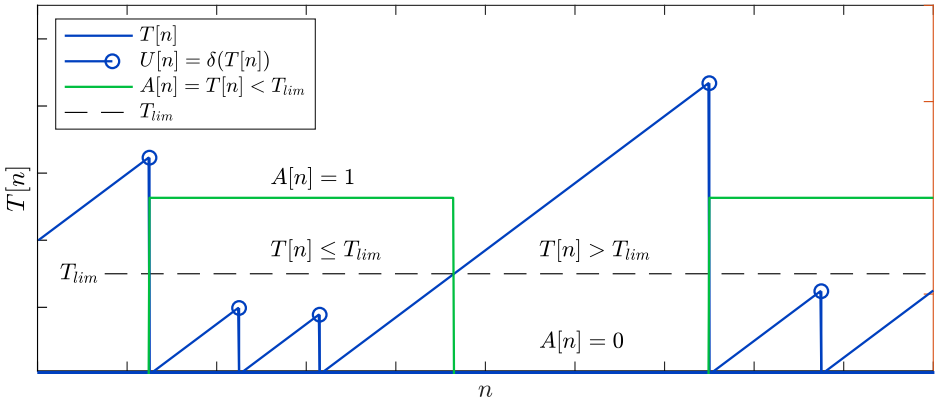


Figure 3.9: Relation between sawtooth sequence $T[n]$, spike train $U[n]$ and activation $A[n]$.

3.3.2 Inter-pulse interval distribution

Inter-pulse intervals (IPI) of MNs are known to be random values concentrated around an average that is inversely proportional to the firing rate. Therefore, PDF of IPI is usually modeled using some unimodal law, such as normal distribution [5]. However, the last does not have a closed-form hazard rate. In this work, we will model IPI PDF as the logistic distribution, which has a simple closed-form hazard function and easily interpretable location and scale parameters. Another eligible choice is discrete Weibull distribution, used in [17]. However, we find it preferable to have both the RT and IPI distributions of the same type.

Logistic distribution is defined on a continuous domain, while the process $T[n]$ is in discrete time. We will suppose that the sampling rate is sufficiently high so that the continuous distribution can be used directly instead of the corresponding probability mass function.

Let us first define the logistic CDF of IPI of i-th MN:

$$S\left(t, \Theta_i, E\right) = \frac{1}{1 + \exp\left[-\frac{1}{\Sigma_i}\left(t - \frac{1}{G_i E + B_i}\right)\right]} \tag{3.18}$$

with the following notation:

- Vector Θ_i is a concatenation of parameters $\Theta_i = \{G_i, B_i, \Sigma_i\}$.
- Term $1/(G_i E + B_i)$ defines the average value of IPI. That is, since firing rate Φ_i of a MN is a function of the intended effect (see Equation 3.2), so is the average IPI, that can be approximated as $1/\Phi_i$.
- Parameter Σ_i regulates the standard deviation of IPI. We suppose that within the low contraction range it can be approximated by a constant value. That is: $\Sigma_i(E) = \Sigma_i = \text{const}$.

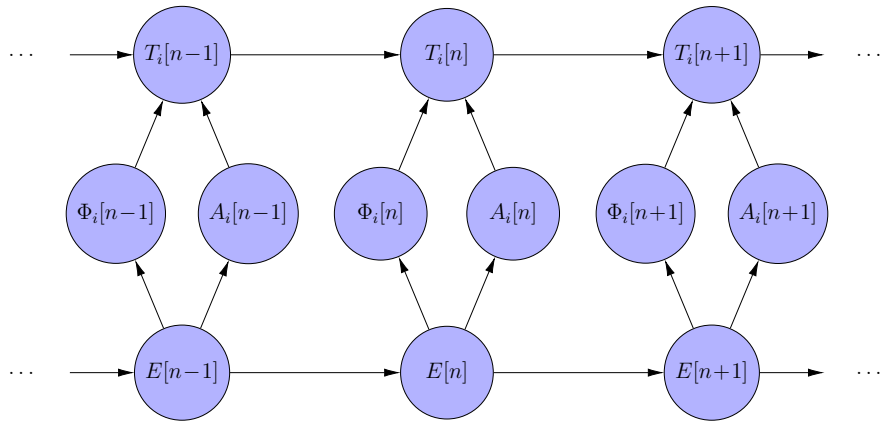


Figure 3.10: Bayesian graph representing the hidden Markov model of MN neuron firing activity. Effect $E[n]$ is a hidden variable that defines the i-th MN's probability of being recruited, via node $A_i[n]$. Additionally, it defines its instantaneous firing rate $\Phi_i[n]$ and, therefore, its IPI distribution, via parameters $\Theta_i[n]$. Sawtooth sequence $T_i[n]$ is an observed value, available from decomposition. At first, we will assume that activation $A_i[n]$ can also be observed using formula (3.16). This assumption will be released later in section 3.3.6

.

Logistic PDF can be easily obtained from its CDF, using the derivative of logistic function.

$$\Pr\left(\Delta_i = t|\Theta_i, E\right) = \frac{1}{\Sigma_i} S\left(t, \Theta_i, E\right)\left(1 - S\left(t, \Theta_i, E\right)\right) \tag{3.19}$$

Corresponding hazard function, by its definition, is:

$$h\left(t, \Theta_i, E\right) \stackrel{\text{def}}{=} \frac{\Pr\left(\Delta_i = t \mid \Theta_i, E\right)}{\Pr\left(\Delta_i \geq t \mid \Theta_i, E\right)} = \frac{1}{\Sigma_i} S\left(t, \Theta_i, E\right) \tag{3.20}$$

3.3.3 Criteria for IPI-based estimation model fitting and inference

Using Markov chain rule, we can write the negative log-likelihood for spike train of i-th MN. Denoting $T^n = T[1..n]$ and $t^n = t[1..n]$:

$$C_i(t_i^n, a_i^n, \Theta_i, E) = -\log \Pr\left(T_i^n = t_i^n \mid \Theta_i, E, A_i^n = a_i^n\right) =$$
$$-\frac{1}{n}\sum_{k=1}^{n} a_i[k] \log \Pr\left(T_i[k] = t_i[k] \mid T_i[k-1] = t_i[k-1], \Theta_i, E\right) \tag{3.21}$$

Similarly to the approach we used to formulate the recruitment-based cost function, we will suppose that spike trains of different MNs are independent. That is, joint negative log-likelihood of spike trains of all MNs is a sum of individual

log-likelihoods:

$$C(t^n, a^n, \Theta, E) = -\log \Pr\left(T^n = t^n \mid \Theta, E, A^n = a^n\right) = \sum_{i=1}^{N} C_i(t_i^n, a_i^n, \Theta_i, E) \quad (3.22)$$

Model identification and effect inference can be realized by optimization of this cost function with respect to parameters Θ or intended effect E:

- *Model identification.* Log-likelihood (3.21) can be used to estimate parameters G, B and Σ that relate effect to the IPI distribution. Similarly to the recruitment-based cost function, (3.21) can be optimized for each MN independently, having a training set of spike trains $\mathcal{D} = \{a_i^D, t_i^D, e^D\}$, where exponent ^{D} denotes all the samples belonging to the training set. That is, the cost function for model identification:

$$C_i^T(\mathcal{D}, \theta_i) = -\log \Pr\left(T_i^D = t_i^D \mid E = e^D,\ A_i^D = a_i^D,\ \Theta_i = \theta_i\right) \quad (3.23)$$

$$\hat{\theta}_i = \underset{\theta_i}{\operatorname{argmin}}\, C_i^T(\mathcal{D}, \theta_i)$$

- *Inference.* Having $t^n[n]$ and $a^n[n]$ observed from decomposition and $\hat{\theta}_i$ estimated in training procedure, we can infer the effect value optimizing (3.22):

$$C^I\left(t^n, a^n, e,\ \hat{\theta}\right) = -\sum_{i}^{N} \log \Pr\left(T_i^n = t_i^n \mid E = e, A_i^n = a_i^n, \Theta_i = \hat{\theta}_i\right) \quad (3.24)$$

$$\hat{e} = \underset{e}{\operatorname{argmin}}\, C^I\left(t^n, a^n, e,\ \hat{\theta}\right)$$

3.3.4 Iterative inference procedure

As we have stated earlier, effect E is essentially a time-varying variable. In order to obtain an estimate that is adequate to the current intent, we will establish a recursive estimation and assign exponentially decaying weights to the decomposition observations. Effect estimation should be performed in real-time, thus we are interested in a fast iterative approach. That is, we will use Newton method [18], which provides the following update of the effect estimate:

$$\hat{e}[n] = \hat{e}[n-1] \ - \ \frac{1}{\ell[n]}\, G^{-1}[n]\, \frac{\partial}{\partial e} C^I\left(t[n], a[n], e, \hat{\theta}\right)\Big|_{\hat{e}[n-1]} \quad (3.25)$$

where $G[n]$ is a smoothed hessian of the inference cost function:

$$G[n] = \left(1 - \frac{1}{\ell[n]}\right) G[n-1] \ + \ \frac{1}{\ell[n]}\, \frac{\partial^2}{\partial e^2} C^I\left(t[n], a[n], e, \hat{\theta}\right)\Big|_{\hat{e}[n-1]} \quad (3.26)$$

and $\ell[n]$ is a time-varying weight that changes according to the following law [18]:

$$\ell[1] = 1, \quad \ell[n] = \left(1 - \frac{1}{\ell_\infty}\right) \ell[n-1] + 1 \tag{3.27}$$

where ℓ_∞ is an equivalent to window length, which can be chosen as 250 ms.

Both the first and second derivatives of the inference cost function can be calculated analytically from (3.24) with respect to effect e. Corresponding expressions are provided in Appendix B.

3.3.5 Example on simulation data

We have simulated spike trains of four MNs using the model described above. More specifically, recruitment thresholds were set to $r = [0.05, 0.3, 0.5, 0.75]$, IPI distribution parameters were $g = [20, 17.5, 15, 12.5]$ Hz, $b = [7, 6, 5, 4]$ Hz, $\sigma = [5, 5, 5, 5]$ ms. Activation was directly simulated as $a_i[n] = e[n] > r_i$, and transition law (3.15) was used to generate spike trains. Effect followed a triangular profile from e=0 to e=1 within 5 s duration.

Simulated spike trains were used to estimate the model parameters by optimization of training cost function (3.23). Obtained values are: $\hat{g} = [20.2, 16.0, 17.0, 12.7]$, $\hat{b} = [6.9, 6.4, 3.9, 4.4]$, $\hat{\sigma} = [5.2, 5.7, 5.7, 6.7]$.

New spike trains were simulated using true distribution parameters for the same effect profile. The estimated model was then used to infer the effect using iterative formula (3.25). Results of estimation are presented in Figure 3.3.5. One can see that the smoothness and precision of the estimate increase with the number of active MNs.

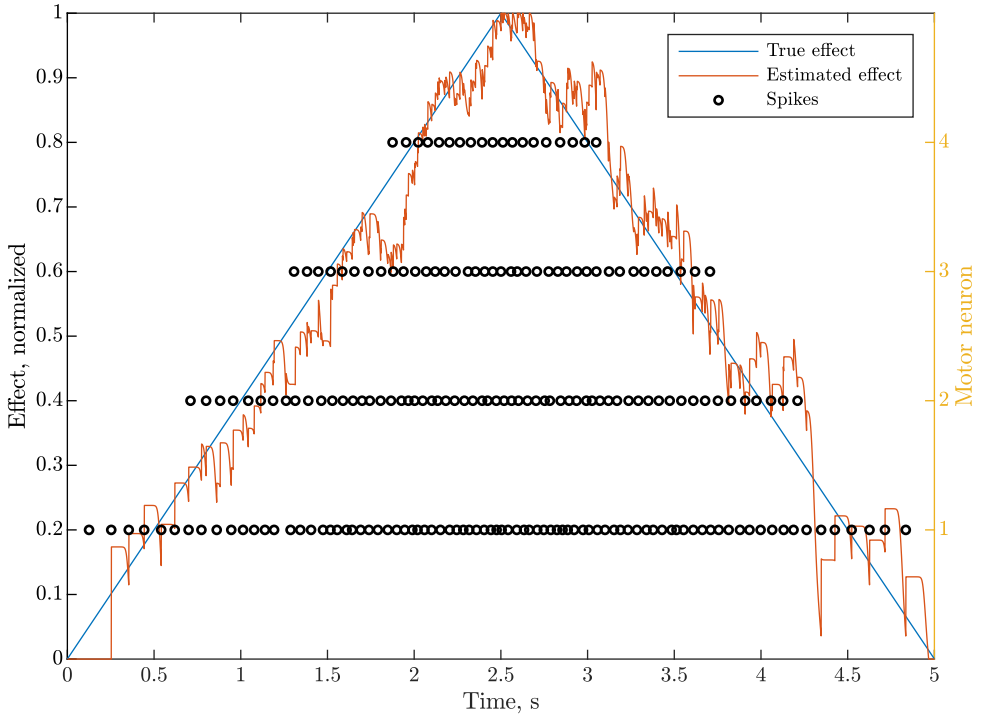


Figure 3.11: Example of IPI-based estimation of effect, on simulated spike trains. Recursive estimation formula (3.25) was used.

3.3.6 Joint effect estimation model based on recruitment and IPI distribution

Previously, we have assumed that activation $A[n]$ can be approximated from observation $T[n]$, using (3.16). However, using total probability rule and conditional distribution of activation (3.4), variable $A[n]$ can be eliminated from the inference. That is, transition law (3.15) can be rewritten the following way:

$$\Pr(T_i[n] = t_i[n] \mid \Theta_i, \Omega_i, E) = \begin{cases} \Pr(A_i[n]=1|E,\Omega_i)\cdot h\left(T_i[n-1],\Theta_i,E\right) & \text{if } t_i[n]=0 \\ \Pr(A_i[n]=1|E,\Omega_i)\cdot\Big(1-h\left(T_i[n-1],\Theta_i,E\right)\Big)+\Pr(A_i[n]=0|E,\Omega_i) & \text{if } t_i[n]=T_i[n-1]+1 \end{cases} \tag{3.28}$$

or, applying conditional law (3.4) and rearranging the terms:

$$\Pr(T_i[n] = t_i[n]|\Theta_i,\Omega_i,E) = \begin{cases} W(E,\Omega_i)\cdot h\left(T_i[n-1],\Theta_i,E\right) & \text{if } t_i[n]=0 \\ 1-W(E,\Omega_i)\cdot h\left(T_i[n-1],\Theta_i,E\right) & \text{if } t_i[n]=T_i[n-1]+1 \end{cases} \tag{3.29}$$

We do not provide an iterative optimization procedure for this cost function. However, to establish the tracking, we can optimize the corresponding log-likelihood cost function in a finite-length window for $k = [n - \ell_w : n]$, where ℓ_w is the window

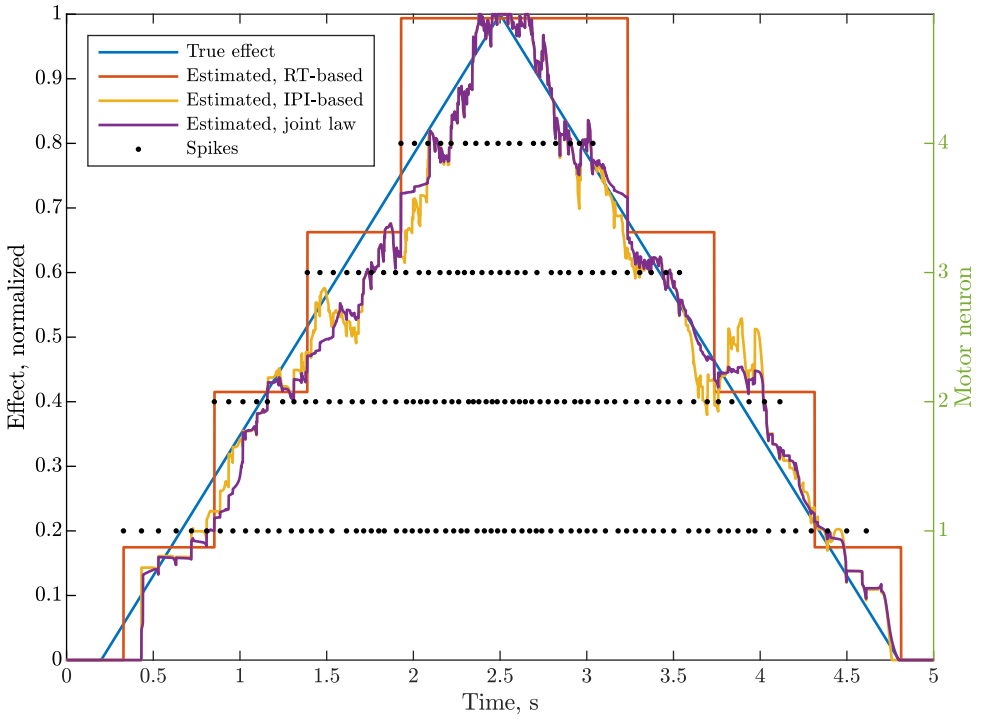


Figure 3.12: Example of joint effect estimation (violet), on simulated spike trains. Also, RT-based (red) and IPI-based (yellow) estimations are provided for comparison.

length. This estimation approach will be used in the following section, as well as in the Results chapter 6.

3.3.7 Example on simulation data

We use the same model parameters as in previous example in Section 3.3.5, that is: $r = [0.05, 0.3, 0.5, 0.75]$, $g = [20, 17.5, 15, 12.5]$ Hz, $b = [7, 6, 5, 4]$ Hz, $\sigma = [5, 5, 5, 5]$ ms. We generate training data using triangular profile with 20 s duration and estimate the model parameters, obtaining: $\hat{g} = [21.0, 16.9, 14.9, 15.0]$, $\hat{b} = [6.5, 6.0, 5.0, 2.0]$, $\hat{\sigma} = [9.4, 10.0, 11.2, 10.0]$, $\hat{\lambda} = [0.04, 0.28, 0.51, 0.74]$, $\hat{k} = [0.03, 0.02, 0.02, 0.02]$.

Figure 3.3.7 shows the results for RT-based (red line, minimization of cost function (3.9)), IPI-based (yellow line, non-recursive minimization of (3.24)) and joint effect estimation (violet line). One can see that joint estimation, compared to IPI-based, exhibits lesser variation. This is due to the regularization introduced by joint transition law (3.28). One can also see that both estimates coincide in the interval where all four MNs are active. This is explained by the fact, that when all MUs are active and the current estimate of effect is larger than last MN's RT, recruitment stops adding any information to the inference, which in this case, starts to be based solely on $T[n]$.

3.4 Conclusion

In this chapter, we have introduced the notion of *effect*, which generalizes the mechanical output of muscle contraction and can represent either a force of contraction, joint torque or joint angle. Then we have assumed, that during movement, the observed effect is equivalent to the intended effect, and therefore can be used to train an intent estimation model. Later, we have introduced a model that links the observed effect with recruitment and instantaneous firing rates of motor neurons.

Next, three effect estimation models based on probabilistic inference were proposed. The first one (recruitment-threshold-based, see Section 3.2) operates solely with the fact that certain motor neurons are recruited or de-recruited, which can be directly observed from decomposition. Due to the discrete nature of its information source, this model provides stepped estimates of the effects. It can be a viable estimation model under the condition that recruitment thresholds of decomposable MNs finely cover the interval of possible effect values. The advantage of this approach is the fact that, instead of a full decomposition, it requires only occasional occurrences of MUAPs that can be detected by, for example, a form-matching algorithm. Also, we have shown that the inference cost function of this model is flat around the optimal value with further fast growth. This defines an interval of reasonable estimates in which the cost function optimum can be easily displaced by additional effect-related information, such as EMG amplitude.

The second estimation model (IPI-based, see Section 3.3.1) uses Hidden Markov modelling, where the intended effect is a hidden variable and sojourn times $T[n]$ of MNs are observations available from decomposition. Transition law of motor neurons' sojourn times is regulated by their instantaneous firing rates, which, in their turn, are linear functions of the intended effect. Parameters of the effect-rate relationship and of the transition law can be estimated from decomposition of a contraction with known measured effect. Then, effect inference from new decomposition data can be performed, using the learned model. A recursive estimation algorithm is proposed to accelerate the inference.

The first two models assume that motor neuron activation can be directly observed from the decomposition data (see (3.16)). Third estimation model (Joint model, see Section 3.3.6) joins the first two by eliminating the activation from the inference, using its conditional distribution with respect to effect (3.4). This permits to regularize the estimate provided by the IPI-based model using the information about recruitment thresholds. Finally, an example of the three models functioning on the same simulated data set is provided (see Figure 3.3.7).

Results on intended effect estimation for data simulated using the model presented in Chapter 5, as well as for experimental data, will be provided in Chapter 6.

Appendix A. Total probability law for the MN recruitment

Total probability law applied to the conditional distribution of activity of i-th MN $\Pr(A_i|E_i,\Omega_i)$, gives:

$$\Pr(A_i = a_i|E,\Omega_i) = \int_0^\infty \Pr(A_i = a_i|E,R_i = r_i)\Pr(R_i = r_i|E,\Omega_i)\,\mathrm{d}r_i \tag{3.30}$$

Using (3.3), we have, for a_i=1:

$$\begin{aligned}\Pr(A_i = 1|E,\Omega_i) &= \int_0^E 1\cdot\Pr(R_i = r_i|E,\Omega_i)\,\mathrm{d}r_i + \int_E^\infty 0\cdot\Pr(R_i = r_i|E,\Omega_i)\,\mathrm{d}r_i \\ &= \int_0^E \Pr(R_i = r_i|E,\Omega_i)\,\mathrm{d}r_i = W(E,\Omega_i)\end{aligned} \tag{3.31}$$

Similarly, for $a_i = 0$:

$$\Pr(A_i = 0|E,\Omega_i) = \int_E^\infty \Pr(R_i = r_i|E,\Omega_i)\,\mathrm{d}r_i = 1 - W(E,\Omega_i) \tag{3.32}$$

Appendix B. Expressions for the first and second derivatives of IPI-based inference cost function

Cost function (3.21) is a sum of, essentially, two types of terms: $-\log\left(h\left(t,\Theta_i,E\right)\right)$ and $-\log\left(1-h\left(t,\Theta_i,E\right)\right)$. In this section we provide the first and second derivatives of these terms relative to the intended effect E. First, let us provide the expression for hazard function $h(\cdot)$:

$$h\left(t,\Theta_i,E\right) = \frac{1}{\sigma_i}S\left(t,\Theta_i,E\right) \tag{3.33}$$

where $\Theta_i = G_i, B_i, \Sigma_i$ and $S(\cdot)$ is a logistic function defined as follows:

$$S\left(t,\Theta_i,E\right) = \frac{1}{1+\exp\left[-\frac{1}{\Sigma_i}\left(t-\frac{1}{G_iE+B_i}\right)\right]} \tag{3.34}$$

For brevity, let us replace the notations $h(t,\Theta_i,E)$ and $S(t,\Theta_i,E)$ by, respectively h and S. First-order derivatives are as follows:

$$\frac{\partial\log h}{\partial E} = \frac{1}{\Sigma_i}\left(1-S\right)\frac{G_i}{(G_iE+B_i)^2} \tag{3.35}$$

$$\frac{\partial\log\left(1-h\right)}{\partial E} = -\frac{h}{1-h}\frac{\partial\log h}{\partial E} \tag{3.36}$$

Second-order derivatives are:

$$\frac{\partial^2 \log h}{\partial E^2} = -\frac{\partial \log h}{\partial E}\left(h + \frac{G_i}{G_i E + B_i}\right) \tag{3.37}$$

$$\frac{\partial^2 \log (1-h)}{\partial E^2} = -\frac{\partial \log h}{\partial E}\frac{h}{1-h}\left(\frac{1}{\Sigma_i}\frac{1-S}{1-h} - \frac{2G_i}{G_i E + B_i} - h\right) \tag{3.38}$$

Since $0 < S < 1$ and $0 < h < 1/\Sigma_i$, one can see that $\partial \log h/\partial E$ is positive, as long as G_i stands positive (negative value would not be physiologically reasonable). Thus, $\partial \log (1-h)/\partial E$ is negative. Second derivative $-\partial^2 \log h/\partial E^2$ is positive as long as $G_i E + B_i$, which equals to a firing rate, returns a positive value. However, $\partial^2 \log (1-h)/\partial E^2$ changes sign. Thus, the corresponding cost function is not convex. However, in practice this does not cause any divergence of the corresponding iterative optimization procedure.

Bibliography

[1] E. Henneman, G. Somjen, and D. O. Carpenter, "Functional significance of cell size in spinal motorneurons," *Journal of Neurophysiology*, vol. 28, pp. 560–580, May 1965.

[2] C. J. De Luca and Z. Erim, "Common drive of motor units in regulation of muscle force," *Trends in Neurosciences*, vol. 17, pp. 299–305, Jan. 1994.

[3] D. Kernell, "The Limits of Firing Frequency in Cat Lumbosacral Motoneurones Possessing Different Time Course of Afterhyperpolarization," *Acta Physiologica Scandinavica*, vol. 65, no. 1-2, pp. 87–100, 1965.

[4] D. Kernell and H. Sjöholm, "Repetitive Impulse Firing: Comparisons between Neurone Models Based on 'Voltage Clamp Equations' and Spinal Motoneurones," *Acta Physiologica Scandinavica*, vol. 87, pp. 40–56, Jan. 1973.

[5] A. J. Fuglevand, D. A. Winter, and A. E. Patla, "Models of recruitment and rate coding organization in motor-unit pools," *Journal of neurophysiology*, vol. 70, no. 6, pp. 2470–2488, 1993.

[6] C. J. De Luca and P. Contessa, "Hierarchical control of motor units in voluntary contractions," *Journal of Neurophysiology*, vol. 107, pp. 178–195, Jan. 2012.

[7] S. J. Garland and T. S. Miles, "Control of motor units in human flexor digitorum profundus under different proprioceptive conditions," *The Journal of Physiology*, vol. 502, pp. 693–701, Aug. 1997.

[8] C. J. De Luca, J. A. Gonzalez-Cueto, P. Bonato, and A. Adam, "Motor Unit Recruitment and Proprioceptive Feedback Decrease the Common Drive," *Journal of Neurophysiology*, vol. 101, pp. 1620–1628, Mar. 2009.

[9] C. J. De Luca and Z. Erim, "Common Drive in Motor Units of a Synergistic Muscle Pair," *Journal of Neurophysiology*, vol. 87, pp. 2200–2204, Apr. 2002.

[10] C. J. De Luca and B. Mambrito, "Voluntary control of motor units in human antagonist muscles: Coactivation and reciprocal activation," *Journal of Neurophysiology*, vol. 58, pp. 525–542, Sept. 1987.

[11] J. L. G. Nielsen, S. Holmgaard, N. Jiang, K. B. Englehart, D. Farina, and P. A. Parker, "Simultaneous and Proportional Force Estimation for Multifunction Myoelectric Prostheses Using Mirrored Bilateral Training," *IEEE Transactions on Biomedical Engineering*, vol. 58, pp. 681–688, Mar. 2011.

[12] A. L. Fougner, Ø. Stavdahl, and P. J. Kyberd, "System training and assessment in simultaneous proportional myoelectric prosthesis control," *Journal of Neuroengineering and Rehabilition*, p. 13, 2014.

[13] P. Romaiguere, J.-P. Vedel, and S. Pagni, "Comparison of fluctuations of motor unit recruitment and de-recruitment thresholds in man," *Experimental Brain Research*, vol. 95, Aug. 1993.

[14] A. Albert and J. A. Anderson, "On the Existence of Maximum Likelihood Estimates in Logistic Regression Models," *Biometrika*, vol. 71, no. 1, pp. 1–10, 1984.

[15] T. Yu, K. Akhmadeev, E. L. Carpentier, Y. Aoustin, and D. Farina, "On-line recursive decomposition of intramuscular EMG signals using GPU-implemented Bayesian filtering," *IEEE Transactions on Biomedical Engineering*, p. 12, 2019.

[16] J. Monsifrot, *Model of Electromyographic Signals by Renewal Processes - Bayes Filter for a Sequential Estimation of Parameters given to the Command of an Upper Limb Prosthesis*. Theses, Université de Nantes, Dec. 2013.

[17] J. Monsifrot, E. Le Carpentier, Y. Aoustin, and D. Farina, "Sequential Decoding of Intramuscular EMG Signals via Estimation of a Markov Model," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, pp. 1030–1040, Sept. 2014.

[18] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*. Massachusetts and London: The MIT Press, 1983.

Chapter 4

Simulation of iEMG

Table 4.1: Main acronyms and notations

MF	Muscle fiber
MU	Motor unit
MNAP	Motor neuron action potential
SFAP	Single fiber action potential
MUAP	Motor unit action potential
NMJ	Neuromuscular junction
FPS	Farthest point sampling
f_s, T_s	Sampling frequency and sampling period of the simulation model
N, F, B, P, D	Respectively, total numbers of MNs (MUs), MFs, axon branches, observation points and trajectory nodes. Also, N is the index of the largest motor neuron
$\cdot_n$, $\cdot_f$, $\cdot_b$, $\cdot_p$	Indices designating a specific MN (MU), MF, axon branch and observation point
F_n, B_n	Total numbers of MFs and axon branches in n-th MU
r_n, s_n, a_n	Recruitment threshold, size and innervation area of n-th motor neuron
$\phi_{fp}(t)$	SFAP of f-th MF in observation point p
$\Phi_{np}(t)$	MUAP of n-th MN in observation point p
$y_p(t)$	EMG signal in observation point p
$\mathbf{Y}^{\mathbf{E}}(t)$	EMG signal in multichannel electrode $\mathbf{E}$; other symbols in bold designate matrices and vectors, instead of scalars

Note: Notation from previous chapter does not apply in this one, except for general terms such as EMG, MU, MN and MUAP, as well as recruitment threshold r.

4.1 Introduction

Intramuscular EMG (iEMG) modeling supports the interpretation of the iEMG signal generation in human muscles. It permits to variate the parameters of both the motor neuron pool and the muscle in order to test and validate iEMG-based computational methods, such as decomposition and motor neuron (MN) territory assessment.

Different applications require simulation models of different complexities. For example, iEMG decomposition algorithms have been often tested using signals simulated by *phenomenological* EMG models [1]. These approaches involve convolution of experimental or simulated spike trains with experimental motor unit action potentials (MUAPs) and provide known spike trains and adjustable level of additive noise. However, they lack modeling of the neuromuscular jitter, morphological MUAPs variability, distant motor units' noise, and electrode geometry.

Compared to the phenomenological approach, *biophysical modelling* of iEMG includes the calculation of each single fiber action potential (SFAP) as a function of the fibers' morphology and of the electrode's relative position. This approach provides an infinitely wide dictionary of MUAPs, permits to vary the sampling rate and may take into consideration the electrode position and the neuromuscular jitter. Moreover, a biophysical EMG simulation can be complemented by a force generation model in order to establish a complete model of the muscle electrical and mechanical responses [2, 3]. In this chapter, we present a new biophysical MUAPs simulation model.

Development of this simulation model pursued two main goals: generation of a decomposable set of motor units and simulation of neural command to the contracting muscle. The modelling approaches to the second goal will be discussed in Chapter 5. The first goal involves modelling of motor neurons' innervation zones, of electrode geometry and of motor unit action potentials. We choose to model these aspects since they define the amplitudes and forms of the MUAPs, as well a the level distant MUs noise. These parameters usually determine the choice of MUs that should be included in the decomposition.

That is, the simulation model proposed in this chapter will address a physiologically correct generation of MUAPs and iEMG with respect to the geometries of MN innervation territories and of the electrode. It will consist of several major sub-models: motor neuron pool, motor unit innervation areas, terminal arborization geometry and jitter, muscle fiber morphology, action potentials generation, and, finally, electrode and noise modeling. As we will see, the established model will also permit to simulate multichannel iEMG, as well as the recordings from electrodes that move during the acquisition process. This result broadens the area of possible applications of this model.

For this purpose, we targeted the simulation of MUAPs recorded by gradually shifting (or scanning) multichannel electrodes, which requires a biophysical approach to simulation, including modelling of the motor unit (MU) territories, neuromuscular junctions (NMJ) locations, and muscle fiber (MF) parameters.

For example, the models described in [4] and [5] provide the simulation of a single

MUAP, not considering the mutual arrangement of MU territories. This problem was later addressed in [6] by using a geometric model of MU territory locations in the muscle cross-section. However, this previous approach assigns an exact number of MFs to each MN, which forces the innervation of MFs at large distances from the imposed territory. While this issue minimally influences the single-channel, localized, intramuscular recordings, we have observed that it has a substantial effect on the multi-channel ones.

Also the unimodal Gaussian and uniform distributions used to generate the NMJ locations in previous approaches [6, 7] produce an excessive scattering of single fiber action potentials (SFAPs) propagation delays. This makes the simulated MUAP waveforms more complex and with greater changes across the channels than in experimental observations [8, 9]. It was suggested in [8] that the axial distribution of the neuromuscular junctions (NMJs) should be instead composed of smaller sub-distributions corresponding to the separate branches of the axon.

In order to address these issues, we propose a new modelling approach that modifies several aspects of the previous modelling works, while preserving some of their properties [6, 10, 4, 11].

In this work, we propose several novel techniques that altogether provide a simulation of physiologically correct MUAPs recorded by intramuscular multi-channel electrodes. These approaches can also be used in combination with previous ones [6, 11, 10, 4] to pursue different simulation goals. More specifically, the new elements we propose include: 1) a new way to generate even distributions of MF coordinates and MU territory centers; 2) a controllable and accurate method for MF-MN assignment; 3) an improved model of MN axon terminal arborization; 4) multi-channel intramuscular electrode modeling; 5) extension to shifting or scanning intramuscular electrode modeling;

Although the proposed model focuses on the simulation of MUAPs, it can be easily extended to a full iEMG generation. This requires an appropriate MN pool model for the generation of spike trains, such as that presented in [7]. The corresponding modeling will be addressed in 5.

The remainder of this chapter is organized as follows. MFs and innervation centers distributions in the muscle cross-section are presented in section 4.2. Next, the fiber-neuron assignment procedure is described in section 4.2.4. Section 4.3.4 explains the modelling of terminal arborizations and of MN action potential (MNAP) propagation delays. Finally, the simulation of MUAPs recorded by multichannel and scanning electrodes is described in Section 4.4.7. Examples that demonstrate the performance of the proposed sub-models will be presented within the corresponding sections.

4.2 Modeling of motor unit territories and muscle fiber innervation

4.2.1 Recruitment thresholds and sizes of motor neurons

It is well known from the Hennemann size principle [12] that motor neurons are recruited and de-recruited in a certain order, depending on their sizes. That is, during increasing contraction the smaller motor neurons are recruited earlier than the larger ones. This is due to the fact that the input resistance of a motor neuron is inversely proportional to its physical size, which makes the smaller motor neurons start firing at lower values of the net excitatory drive. These values are called *motor neuron's recruitment thresholds*.

The authors of [7] have proposed a model of recruitment thresholds distribution across the motor neurons of a muscle. According to their model, thresholds are distributed exponentially, so that the number of early-recruited motor units is larger compared to the later-recruited ones, which reflects the distributions obtained in the experiments [12].

We suppose that the excitatory input to the motor neuron pool is normalized to fit the interval between 0 and 1, so that the input corresponding to maximum voluntary contraction is equal to 1. Thus, the recruitment thresholds (RTs) of MNs will be measured as the value of excitatory input at which the MN is recruited (i.e., starts to fire). The sizes of MNs will be considered proportional to the recruitment thresholds and normalized to the same interval.

We define the distribution of MN RTs, similarly to the one proposed in [7]:

$$r_n = k \exp\left(a[n-1]\right) \tag{4.1}$$

where n denotes the $n-th$ motor neuron, while a and k are respectively slope and normalization parameters.

In order to identify a, one applies the constraint $r_N/r_1 = R$, where R denotes the magnitude of the RT distribution (RT of the largest MN related to RT of the smallest one). This gives $a = \ln(R)/(N-1)$. Normalization parameter k is identified by constraining $r_N = M$, where $M \leq 1$ is the maximum RT in the pool, which gives:

$$\frac{1}{k} = \frac{1}{M} \exp\left(\frac{N-1}{N-1} \ln R\right) = \frac{R}{M}$$

So that the final equation for RT distribution:

$$r_n = \frac{R}{M} \exp\left(\frac{n-1}{N-1} \ln R\right) \tag{4.2}$$

We suppose that the MN sizes are proportional to their RTs and are bound to the interval $(0, 1]$, so that:

$$s_n = r_n/r_N \tag{4.3}$$

Units of MN size s and RT r are arbitrary, only their relative magnitudes will be used in further calculations. These are the key parameters to the following parts of the model since they define the motor neurons' firing rates across the entire activation range and motor units' innervation areas.

4.2.2 Distribution of the innervation centers in the muscle cross-section

Following the approach used in previous studies, [6, 11, 13], we approximate the muscle by a cylinder where all MFs are parallel to the z-axis and where the xy plane constitutes the muscle cross-section (see Figure 4.1). In order to assign MFs to MNs and simulate MUAPs, the territories of the MUs in the muscle cross-section should be first defined. A MU territory is a circular region in the muscle cross-section determined by two parameters: the xy coordinates of its center and its area. Anatomically, it is the area that includes all the MFs innervated by the MN. In this section, we describe a new method to generate the territory centers, while the areas will be addressed later in Section 4.2.4.

In the existing models [7, 11], the coordinates of MN territory centers are assumed to evenly fill the muscle cross-section. For this purpose, the territory centers are drawn from a uniform distribution over the muscle cross-section. However, the uniform distribution provides very uneven arrangement of the points (see Figure 4.2 for illustration). In turn, this leads to uneven and physiologically incorrect densities of the innervated MFs, as shown in [14]. Another simulation model [6] generates centers as randomly altered nodes of a rectangular grid, which partly resolves this problem, but may be inconvenient to implement due to the fact that the generated centers may end up outside of the admissible region.

We propose an alternative approach that consists of using the farthest point sampling (FPS) [15, 16]. FPS is a family of algorithms that fill an enclosed 2D domain by iteratively adding points that are maximally distant from the previously added ones. This property of the FPS algorithms allows to maximally disperse the MU territory centers in the muscle cross-section, thus achieving an even scattering. At the same time, the generated points are quasi-random (e.g., their arrangement solely depends on the form of the region and on the position of the initial point) and do not follow any regular pattern, such as the rectangular grid. A similar approach was proposed in [14], where the Mitchell's Best Candidate method was applied [17].

FPS algorithms provide an additional feature that may be of interest in iEMG simulation. They place each new territory center as far as possible from the previously generated ones. The subsequent assignment of the territory centers to the MUs can be performed either randomly or in a specific order. If it is performed in the order of decreasing MN sizes, the centers of the MUs with similar sizes will be as distant from each other as possible, thus achieving an even distribution of MUs of different sizes in the cross-section. An illustration is provided in Figure 4.3, where the circles around the territory centers indicate the relative sizes of corresponding MUs. In this example, the muscle cross-section is evenly filled with territory centers of MUs of different sizes. This property is relevant when an even distribution of size-

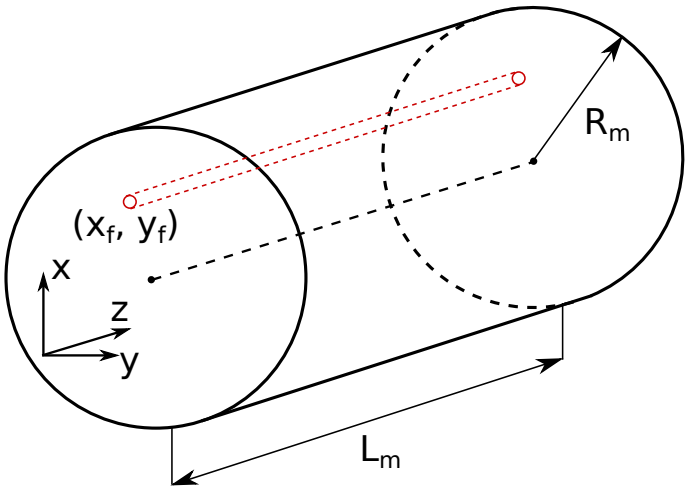


Figure 4.1: Geometrical model of the muscle: a cylinder with radius R_m and height L_m; muscle fibers (red), are straight parallel lines characterized by their position in the muscle's cross-section (x_f, y_f).

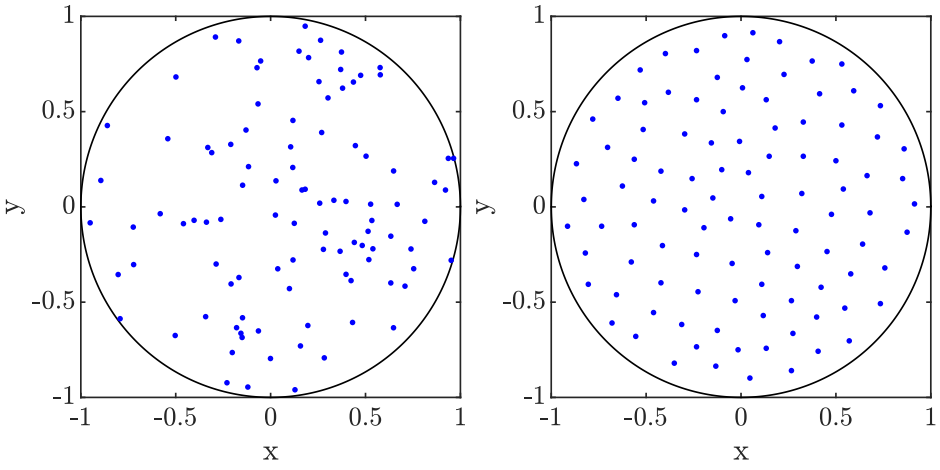
dependent parameters in the muscle cross section is of interest. Possible examples are the distributions of MFs of types I and II [18] (types I and II are predominant in, respectively, smaller and larger MUs [12]), as well as MF diameters [19, 20] (MFs of type I tend to have smaller diameters than those of type II [21]). Even distribution of MUs of different sizes will contribute to an even distribution of such parameters in the muscle cross section. Alternatively, MNs can be assigned to the territory centers in a randomized order if the aforementioned structure is not required.

4.2.3 Muscle fibers geometry and distribution in the muscle cross section

In this model, similarly to [6, 13, 22], MFs are modeled as straight parallel lines with their extremities located in the tendon regions, while the muscle is modeled as a cylinder (see Figure 4.1).

The distribution of MFs in the muscle cross-section (xy-plane) should be uniform and, preferably, should take into consideration the diameters of MFs. In previous works [7, 11], the MFs' locations were drawn from the uniform distribution within the corresponding MU territory. As it was noted above, this method, combined with previous approaches to the territory centers generation, does not guarantee the global uniformity of MF density in the entire muscle cross-section. Alternatively, in [6] MFs were positioned prior to the generation of the territory centers in the nodes of a regular rectangular grid.

We propose to generate the MFs locations using the aforementioned FPS algorithms family. In our simulation model, we generate them independently from the territory centers, and simulate the MF-MN assignment in following steps, similarly to [6]. Figure 4.2 provides a comparison of the locations generated by this method with those drawn from the constant distribution. In our simulation we use constant



(a) Independent samples from the uniform distribution

(b) Farthest point sampling

Figure 4.2: Example of 100 points (territory centers or MF locations) drawn from the uniform distribution (a) and generated by the FPS (b). An even quasi-random arrangement of the points, such as the one provided by the FPS, is not achievable when using the uniform distribution.

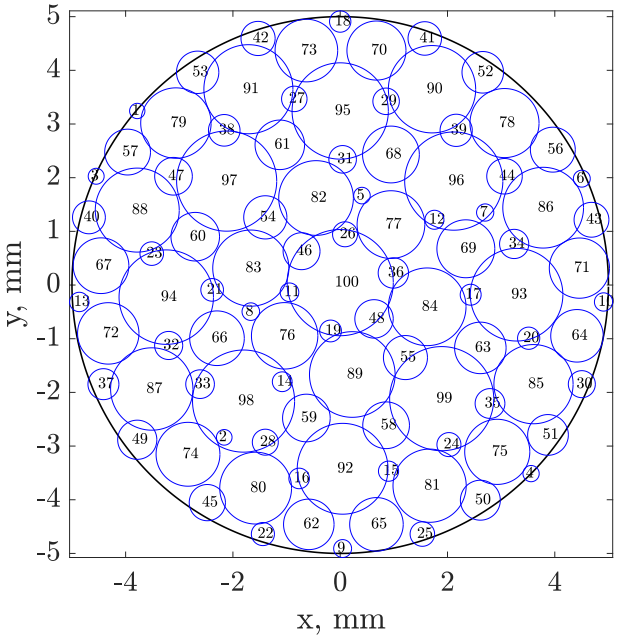


Figure 4.3: Territory centers of N=100 MN (magnitude R=50) across the cross-section area of the muscle; blue circles are drawn to illustrate the relative sizes of the corresponding MUs (and not their territories). The FPS method allows an even distribution of not only the centers, but also of the sizes of MUs.

density of 400 MFs per mm^2. However, it is worth mentioning that the FPS also allows a variable local density of MFs.

4.2.4 Assignment of muscle fibers to the motor neurons

We establish a randomized procedure in which fibers are assigned to motor neurons according to several physiological factors: the expected number of fibers innervated by the neuron, fiber's proximity to the innervation center and presence of neighboring fibers already assigned to that motor neuron. For each neuron-fiber pair, the probability of assignment is represented by a *score* that combines influences of each factor:

$$P_f(n) \sim P_n^a \cdot P_n^g(x_f, y_f) \cdot P^d(n, f, n_c) \tag{4.4}$$

where

- $P_f(n)$ is the score that characterizes the probability of the f-th MF to be assigned to the n-th MN, given the size of the MN, positions of the MF and MN's territory center, and neighboring MFs innervation;
- P_n^a denotes the *a priori* probability of assignment, i.e., solely given the size of the MN;
- $P_n^g(x_f, y_f)$ denotes the probability of MF with coordinates (x_f, y_f) to be innervated by n-th MN, given its territory center location;
- $P_n^d(n, f, n_c)$ is an indicator function returning 0 if any of n_c closest neighbors of f-th MF is already assigned to n-th unit, and 1 otherwise.

The pseudocode of the fiber assignment algorithm is presented in Algorithm 1. Let us consider each multiplier in (4.4) in a detailed way.

A priori probability of assignment

A priori probability of an MF being innervated by n-th MN is proportional to the total number of MFs that this MN should innervate. We suppose that this number is itself proportional to the size of the MN. Thus, we calculate the *a priori* probability P_n^a of MF innervation in the following way:

$$P_n^a = \frac{s_n}{\sum_{n=1}^{N} s_n} \tag{4.5}$$

where s_n is the size of the n-th MN which can be modeled using an exponential distribution for recruitment thresholds, as proposed in [7].

Distribution of MF coordinates around an innervation center

We assume that the innervation territories of MNs are circular; this assumption is common to most EMG simulation models and is supported by experimental data [23]. We model the P_n^g as a symmetrical two-dimensional Gaussian distribution:

$$P_n^g(x,y) = \frac{1}{S_n} \cdot \frac{1}{2\pi\sigma_n^g} \exp\left[-\frac{(x-\mu_{nx}^g)^2 + (y-\mu_{ny}^g)^2}{2\sigma_n^{g\,2}}\right] \tag{4.6}$$

where x and y are the coordinates of the MF; mean μ_i^g is coincident with the territory center of n-th MN; S_n is an out-of-border coefficient (see explanation below in this section); standard deviation $\sigma_n^g = a_n/\pi C$ is proportional to the MN's innervation area a_n with scattering coefficient C.

We assume that the innervation areas of MNs are proportional to their sizes with a scaling factor A/k: $a_n = s_n/s_N \cdot A/k$, where A is the area of the muscle cross-section. The value of k sets up the area of the largest MN as a fraction of the muscle cross-section area: $k = A/a_N$. The value of k varies across muscles. In our simulation, we have chosen a value of k=4. The value of the scattering coefficient C regulates the tightness of the Gaussian distribution of MFs around the territory center. We calculate it assuming that a_n is the area of 0.99 confidence circle for the corresponding distribution, giving us $C = \text{inv-}\chi^2(0.99, 2) = 9.21$.

Innervation areas of some MNs may partly lie outside of the muscle border. Coefficient S_n takes this fact into consideration and normalizes the corresponding distribution (4.6) in order to ensure that the number of innervated MFs will still be correct. S_n is calculated as a double integral of the original Gaussian distribution in (4.6) above the domain corresponding to the muscle cross-section. Thus, the probability P_n^g sums to 1 while integrating over the muscle region. Considering the classification proposed in [19], this approach can be assigned to *uniform-augmented* territory placement.

Adjacency of fibers innervated by the same motor neuron

Due to the phenomena of self-avoiding in the arborizations of MNs axons, MFs of the same motor unit rarely lie next to each other. This fact is reflected in eq. (4.4) by using an additional factor $P^d(n, f, n_c)$, which equals to zero if at least one of n_c closest fibers is already innervated by n-th motor neuron. This principle is also in accord with the maps of glycogen-depleted muscle fibers belonging to a single motor neuron, obtained in the experiments [24].

The value of n_c also regulates the scattering of MNs' fibers across the muscle cross-section. We suggest $n_c = 5$ for a regular modeling strategy. This choice prevents the formation of dense clusters of fibers while authorizing a limited adjacency.

4.2.5 Results of the MFs innervation modeling

In order to demonstrate the performance of the fiber innervation model, we simulated a muscle of 10 mm in diameter with a mean fiber density of 400/mm^2, resulting in

Algorithm 1 Fiber assignment procedure

while not all fibers are assigned **do**
  w = zeros(N,1);
  f = random non-assigned fiber;
  for each motor neuron n **do**
    $w_n \leftarrow P_n^a \cdot P_n^g(x_f, y_f) \cdot P^d(n, f, n_c)$;
  end for
  assign fiber f to a random neuron n with weight w_n;
end while

approximately F=34000 fibers. It was innervated by N=100 motor neurons with size magnitude R=50. The resulting numbers F_n of innervated fibers per motor neuron are shown in Figure 4.4. This distribution follows very closely the one imposed by the model $F_n = F \cdot s_n / \sum_{n=1}^{N} s_n$.

The resulting innervation areas also lie close to their model-imposed values. They can be calculated both as areas of convex hulls or areas of 0.99 confidence ellipsoids. An example of resulting innervation territories' forms is provided in Figure 4.5.

4.2.6 MF diameters and conduction velocities modeling

The diameter of a MF defines its conduction velocity and transmembrane potential current density (see Eq. (4.12)). We simulate the diameters following the procedure described in previous model [6]. This approach seeks to keep the global distribution of diameters to be normal with mean and standard deviation of respectively 55 µm and 9 µm, as observed in experiments. At the same time, it assigns larger diameters to the MFs of larger MNs, as it is also supported by experimental data.

Resulting distribution of MF diameters, generated using this procedure and innervation numbers obtained from our assignment algorithm, is illustrated in Figure 4.6. Resulting histogram closely fits the experimental distribution, which signifies that the two simulation models (previous and the presented one) interact correctly.

MF conduction velocities (MFCVs) are proportional to the fiber diameters. According to [4], MFCVs can be simulated by a linear mapping of the experimental diameter range to the velocities range. That is, we use their parameters that establish a linear mapping from 22-85 µm to 2.2-5.5 m/s, directly giving the values of MFCVs from MF diameters.

4.3 Neuromuscular junction modeling

The neuromuscular junction is the biological interface between a MF and the axon innervating motor neuron axon. In order to innervate all its fibers, an axon splits into smaller branches, forming a complex and uneven tree structure with neuromuscular junctions at its leaves [25].

Motor neuron action potential (MNAPs) originates in the soma of the MN and propagates along the MN axon branches until reaching each of the innervated MFs.

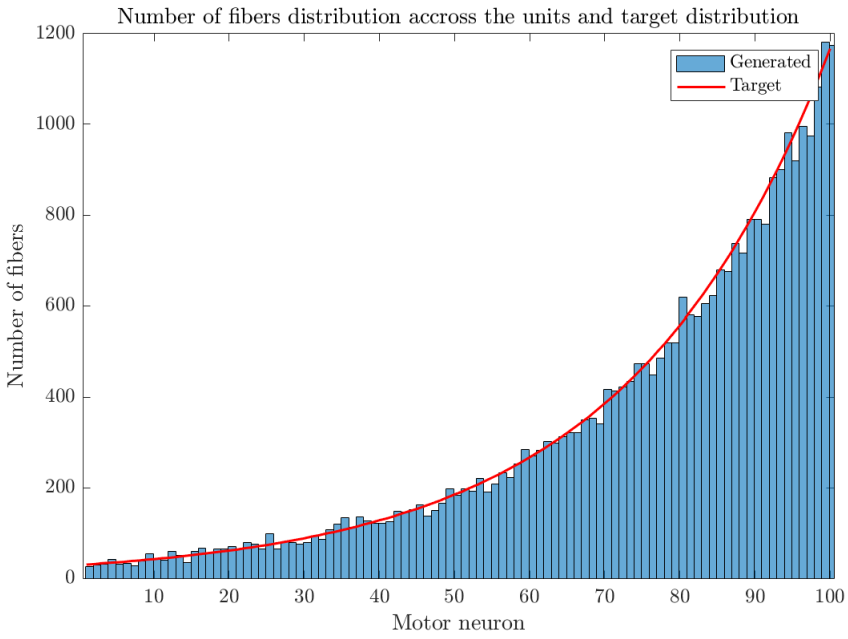


Figure 4.4: The number of fibers assigned to each motor neuron for a simulated muscle with radius of 5 mm, 31400 fibers, N=100 neurons and R=50. Red curve: model, according to motor neuron sizes; bar plot: proposed assignment procedure.

The lengths of the paths to each fiber vary due to the scattering of the neuromuscular junctions in the muscle. This causes the scattering of MNAP propagation delays, which affects the morphology of the MUAPs.

In this section, we will show how our simulation model calculates the coordinates of neuromuscular junctions and MNAP propagation delays.

4.3.1 Structure of the axon branching model

A MN axon can be represented as a root of a tree structure that splits into several branches of smaller radii. This process is then repeated several times within each branch until each muscle fiber is reached.

In our model. we suppose that split is done only twice (see Figure 4.7). Thus, each muscle fiber and its neuromuscular junction are assigned not only to a motor neuron but to a specific branch of its axon. We establish such a model in order to constrain the complexity of MUAPs while providing physiologically correct distributions of NMJs along the muscles.

4.3.2 Fiber-branch assignment

We assume that the number of branches is proportional to the MU's size and that its value defines the number of phases in its action potential. The following expression provides the numbers of branches/phases that correspond to the experimental action

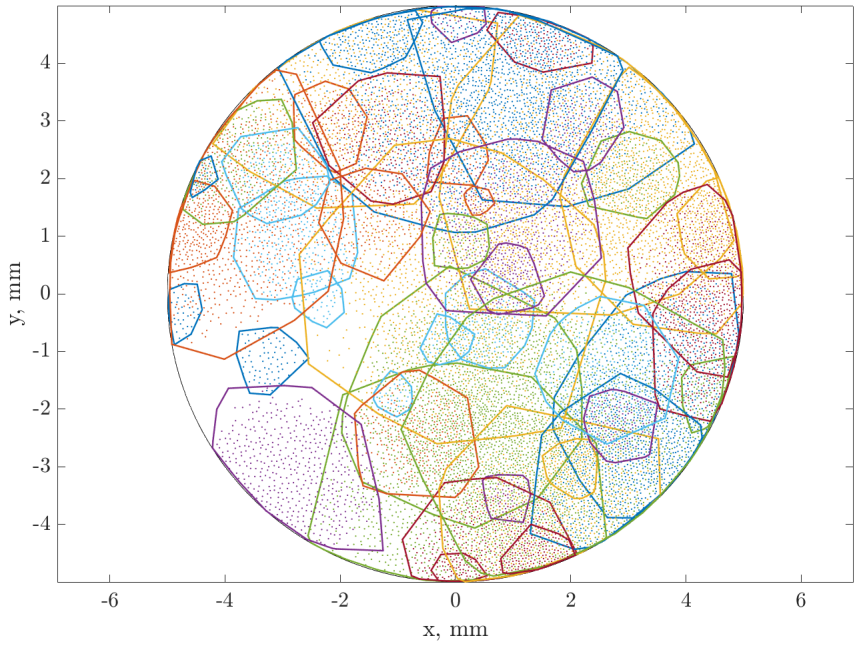


Figure 4.5: Simulated innervation territories of MNs from the same simulation as in Figure 4.4, drawn as convex hulls of innervated fibers coordinates; a random subset of 50 out of 100 MN is shown to avoid clutter. Points of the same color as borders represent the innervated MFs' locations.

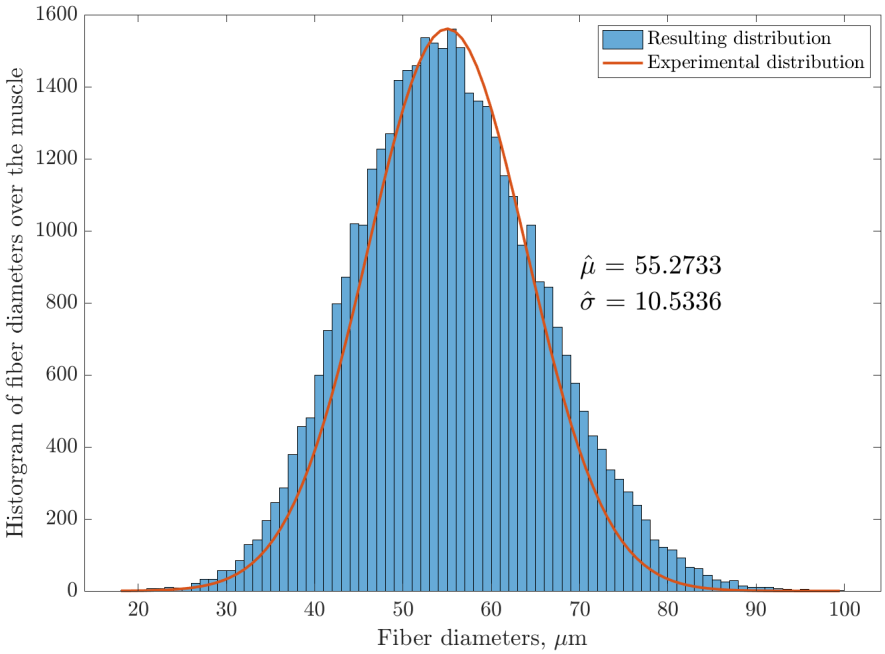


Figure 4.6: Histogram of simulated fiber diameters. Diameters are simulated in the function of the size of innervating MN. Here, the total histogram across the muscle is provided, overlaid with the typical experimental distribution.

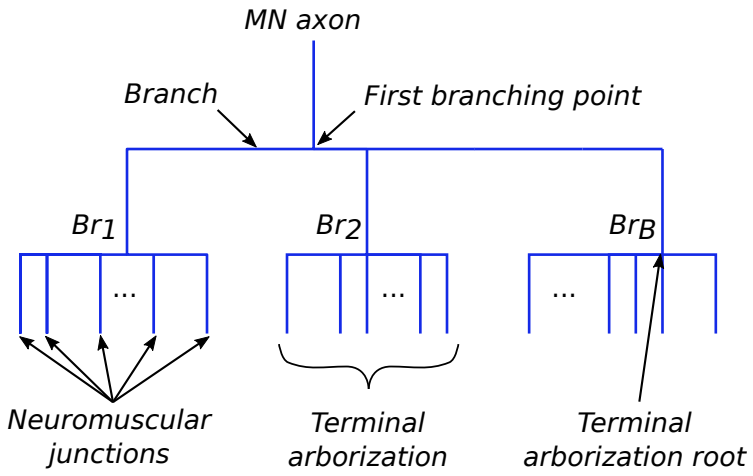


Figure 4.7: Structure of motor neuron axon branching, modeled as a tree with a single bifurcation, where the axon is the root and neuromuscular junctions are the leaves. Leaves are organized into terminal arborizations Br_i using k-means clustering of fiber positions in the cross-sectional plane.

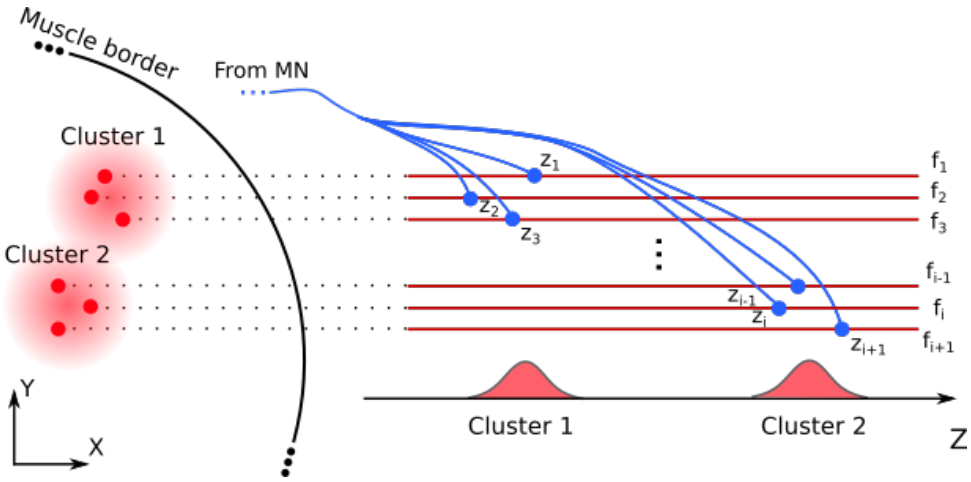


Figure 4.8: Neuromuscular junction z-coordinates distribution model: combination of Gaussians scattered across end-plate zone, each associated with a cluster of MFs in the muscle cross-section.

potentials for small motor units (1-2 phases) as well as for the largest ones (4-6 phases):

$$B_n = 1 + \lfloor \ln(s_n/s_1) \rceil \tag{4.7}$$

where s_n is size of n-th motor unit and $\lfloor \cdot \rceil$ stands for rounding to the nearest integer.

For each motor unit, in order to assign each MF to a specific branch, we first define the number B_n of branches using equation (4.7) and then run the *k-means* clustering algorithm over the motor unit's fibers coordinates in the cross-sectional plane, looking for B_n clusters. In this case *k-means* seeks for B_n groups of closely-located MFs of a MU. Then the MFs of each group are assigned to a single axon branch.

4.3.3 Coordinates of neuromuscular junctions

Neuromuscular junctions coordinates in cross-sectional plane (x, y) are the same as their fibers' coordinates. The assignment of the MFs to specific branches permits to generate a multimodal distribution of the neuromuscular junctions along the z-axis (Figure 4.8). Such distribution has an advantage over previously used uniform or unimodal Gaussian [6] in terms of MUAP waveforms, especially in case of large motor units, where unimodal distributions tend to excessively scatter the SFAP delays, leading to the generation of abnormally polyphasic MUAPs.

Figure 4.8 demonstrates the distribution of MFs of two different branches in the cross-sectional area (left), as well as the densities of their neuromuscular junctions along the z-axis (right). These densities are modeled as Gaussian clusters with mean values scattered across the z-axis of the muscle and standard deviations much smaller than the muscle length.

We calculate mean value μ_n^b and standard deviations σ_n^b of intra-cluster densities using the following model:

$$\begin{aligned}\mu_n^b &\sim g\left(L/2,\ \sigma_n\right),\ \sigma_n = a_\mu + b_\mu \cdot \sum_{k=1}^{n} s_k / \sum_{k=1}^{N} s_k \\ \sigma_n^b &= a_\sigma + b_\sigma \cdot \sum_{k=1}^{n} s_k / \sum_{k=1}^{N} s_k\end{aligned} \tag{4.8}$$

where parameters a_μ, b_μ, a_σ, b_σ in combination with fiber and axon conduction velocities define the dispersion of the MNAP propagation delays, and, thus, the durations of MUAPs.

In order to obtain an initial estimate of these parameters, we impose the largest and the smallest MUs to have MUAPs with durations of 2.5 ms and 7.5 ms respectively, in a multichannel setup, where the electrodes cover the entire area of the MU. Considering mean conduction velocities of their fibers to be 2500 mm/s and 5000 mm/s [4], we can approximately calculate the necessary span of their neuromuscular junctions, giving correspondingly $l_{min} = 6.25$ mm and $l_{max} = 37.5$ mm. We also assume that the standard deviation of the cluster centers σ_n is larger than the intra-cluster deviation σ_n^b, since a MUAP usually contains several distinct phases. In the simulation, we have found it convenient to set $d = \sigma_n/\sigma_n^b$ to 4. Finally, we note that the span of neuromuscular junctions locations along the z-axis for n-th motor unit can be roughly calculated as $3(\sigma_n + \sigma_n^b)$.

These considerations give us the following system of equations:

$$\begin{cases} 3(a_\mu + a_\sigma) = l_{min}, \\ 3(a_\mu + a_\sigma + b_\mu + b_\sigma) = l_{max}, \\ b_\mu / b_\sigma = d, \\ a_\mu / a_\sigma = d; \end{cases} \tag{4.9}$$

solution of which for $d = 4$ gives $a_\sigma = 0.4$, $b_\sigma = 2.1$, $a_\mu = 1.7$, $b_\mu = 8.3$ (all in millimeters). We should consider these values as upper estimates, since MUAPs' lengths are also influenced by MNAP propagation delays (see Section 4.3.4 for details) which were not yet taken into account. According to our observations and for MNAP propagation delays listed in Section 4.3.4, a set $a_\sigma = 0.25$, $b_\sigma = 1$, $a_\mu = 1$, $b_\mu = 2.5$ produces MUAPs with physiologically correct forms and lengths.

4.3.4 Delay of MNAP propagation

Once all the MFs of a motor unit are assigned to their branches and the z-coordinates of NMJs are generated, we can calculate the delays of MNAP propagation towards each junction. We divide the lengths of each segment of the axon (see Figure 4.7) by their propagation velocities, thus, the delay for f-th fiber assigned to b-th branch of n-th motor unit is:

$$d_f = \frac{|\boldsymbol{x}_n^c - \boldsymbol{x}_k^b|}{v_b} + \frac{|\boldsymbol{x}_k^b - \boldsymbol{x}_f^j|}{v_t} \tag{4.10}$$

where

- $\boldsymbol{x}_f^j$ are coordinates of the neuromuscular junction of f-th fiber in k-th branch of n-th motor unit.

- $\boldsymbol{x}_k^b$ are coordinates of the terminal arborization root at the k-th branch of the n-th motor neuron's axon, calculated as the mean of positions of the neuromuscular junctions:

$$\boldsymbol{x}_k^b = \sum_{f=1}^{N_k} \boldsymbol{x}_f^j$$

- $\boldsymbol{x}_n^c$ are coordinates of the first branching point of n-th motor unit, which is calculated as the mean of arborization roots:

$$\boldsymbol{x}_n^c = \sum_{k=1}^{K_n} \boldsymbol{x}_k^b$$

- v_b is MNAP propagation velocity in a branch of the motor neuron axon;

- v_t is MNAP propagation velocity in a terminal arborization of the motor neuron axon;

We assume that branches' propagation velocities are much smaller than that of the axon due to their smaller diameter and the absence of myelination in case of terminal arborization. Values that we used in our model are: $v_b = 10$ m/s, $v_t = 1$ m/s (for comparison, typical propagation velocity of an MN axon is 50 m/s). To our best knowledge, there is yet no experimental data on v_t and v_b in the literature.

4.4 MUAP and EMG modeling for multichannel and shifting electrodes

The generation of muscle fiber and motor unit action potentials is a complex bioelectrical process. In order to establish the simulation of SFAPs and MUAPs, we have studied its available models [26, 27, 4, 28, 10]. We did not alter these models, and in our simulation approach, we have adapted them as they are. For the consistency of our method description, these models will be provided in the two following sections.

The arrival of motor neuron action potential to the neuromuscular junction of a muscle fiber causes the *depolarization* of its cells positioned right next to the junction. This process provokes the depolarization of cells lying next to the first ones, and so on, until this chain reaction reaches the tendon regions of the muscle. Depolarization wave propagates along the muscle fiber with a constant speed of 2.5-5.0 m/s, depending on its diameter.

Depolarization of the muscle fiber consists of a short alteration of the electric potential difference between the inside and the outside of the fiber, called *transmembrane* potential. This potential difference induces a distribution of so-called *transmembrane current sources* along the muscle fiber [10]. These elementary sources

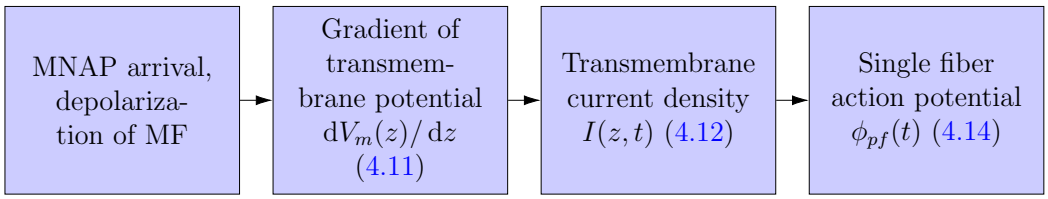


Figure 4.9: Scheme of the SFAP generation model adopted in the simulation.

are considered to form the electric potential field that, measured in the vicinity of the fiber against a neutral point, form the electromyographic signal. We have schematically illustrated this process (see Figure 4.9) in order to simplify the reader's navigation through the following sections.

4.4.1 Transmembrane potential and current density

Transmembrane current density is usually modeled as being proportional to second derivative of transmembrane potential [26] along z-axis, which has the following form:

$$V_m(z) = \begin{cases} Az^3 \exp(-z) + B, & z > 0 \\ 0, & z \leq 0 \end{cases} \tag{4.11}$$

where A is a proportionality coefficient, equal to 96mV$\cdot$ mm^{-3}, and B is potential at rest, equal to -90 mV.

As it was proposed in [10], transmembrane current density $I_f(z,t)$ of f-th muscle fiber can be modeled in time-space domain as follows:

$$I_f(z,t) = d_f \frac{\partial}{\partial z}[\psi(z - z_{0f} - v_f t) \cdot p_{L_f}(z - z_{0f} - L_f/2) - \\ - \psi(-z + z_{0f} - v_f t) \cdot p_{R_f}(z - z_{0f} - R_f/2)] \tag{4.12}$$

where

- d_f is the diameter of the muscle fiber; values of the diameters are modeled as proposed in [6];
- z_{0f} is position of f-th fiber's neuromuscular junction;
- L_f and R_f are distances from the neuromuscular junction of f-th fiber to the left and right tendon regions respectively;
- $\psi(z) = dV_m(z)/dz$;
- $p_{L_f}(z)$ and $p_{R_f}(z)$ are indicating functions equal to 1 if, respectively, $-L_f/2 \leq z \leq L_f/2$ and $-R_f/2 \leq z \leq R_f/2$;
- v_f is fiber's conduction velocity.

4.4.2 Single fiber action potential modelling

In order to simulate single fiber action potentials (SFAPs) we define a potential induced in an observation point $p = [x_p, y_p, z_p]^\top$ by an elementary current source located at a narrow fiber's slice at coordinate z [27, 4]:

$$\phi_{fp} = \frac{1}{4\kappa_r} \cdot \frac{I_e}{\sqrt{r_{fp}^2 \kappa_r / \kappa_z + (z_p - z)^2}} = h_{fp}(z_p - z) I_e \tag{4.13}$$

where

- I_e is an elementary current source;
- z and z_p are locations of the elementary current source and of the observation point respectively;
- r_{fp} is radial distance between muscle fiber f and observation point p, that is, $r_{fp} = \sqrt{(x_p - x_f)^2 + (y_p - y_f)^2}$;
- κ_r and κ_z are radial and axial conductivities of the muscle tissue (0.063 S/m (Siemens per meter) and 0.33 S/m respectively [4]);

As we noted earlier, during contraction, transmembrane current sources are continuously distributed along the fiber, according to (4.12). Individual potential $\phi_{fp}(t)$ of fiber f at the observation point p, generated by this distribution can be calculated by integration [4]:

$$\phi_{fp}(t) = \int_{-\infty}^{\infty} h_f(z_p - z) I_f(z, t)\, \mathrm{d}z \tag{4.14}$$

That is, the single fiber action potential is a convolution of transmembrane current density distribution and elementary current potential. While the former is fiber-specific, the latter is determined by the relative position of the fiber and of the observation point.

4.4.3 Motor unit action potential modelling

The action potential of a MU is modeled as a linear sum of its muscle fibers' SFAPs [6]. Taking into consideration the MNAP propagation delays and neuromuscular jitter [29], a MUAP can be calculated the following way (see also Figure 4.10 for illustration):

$$\Phi_{np}(t) = \sum_{f=1}^{F_n} \phi_{fp}(t - d_f - \zeta) \tag{4.15}$$

where $\Phi_{np}(t)$ is the MUAP of n-th motor unit, observed in point p; F_n is the number of fibers innervated by this motor unit; $\phi_{fp}(t)$ is SFAP of f-th muscle fiber of n-th motor unit, observed in point p; d_f is the delay between the MNAP discharge and

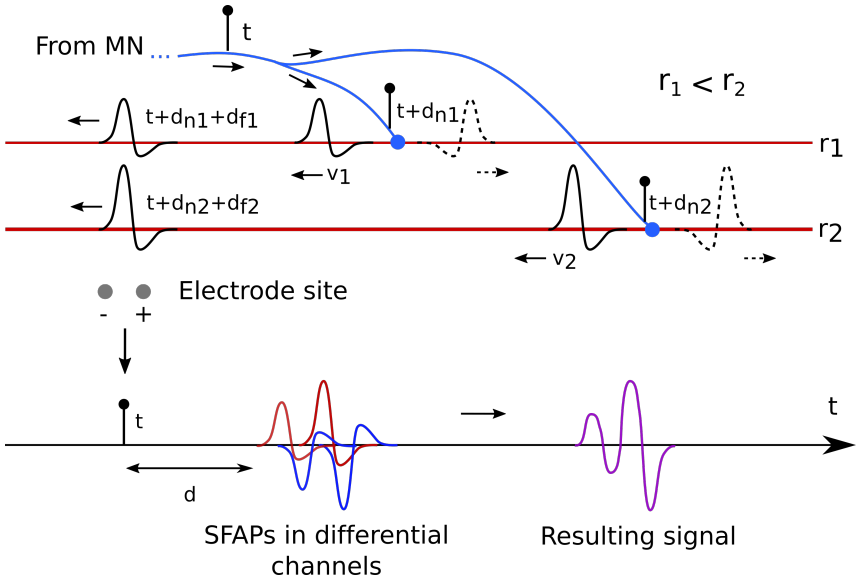


Figure 4.10: Effect of propagation delays of MNAPs (d_n) and SFAPs (d_f) on resulting motor unit action potential.

its arrival to the neuromuscular junction of the fiber (see expression (4.10)); ζ is a delay caused by neuromuscular jitter.

Equations (4.14) and (4.15) operate with continuous arguments and therefore permit to calculate the MUAPs and EMG with infinite temporal precision. Practically, in order to accelerate the simulation, we pre-calculate $\phi_{fp}(t_n)$ for each discrete time value $t_n = \{0, T_s, \ldots, (N-1)T_s\}$, where T_s is sampling period and is N sufficiently big for t_n to cover the entire action potential.

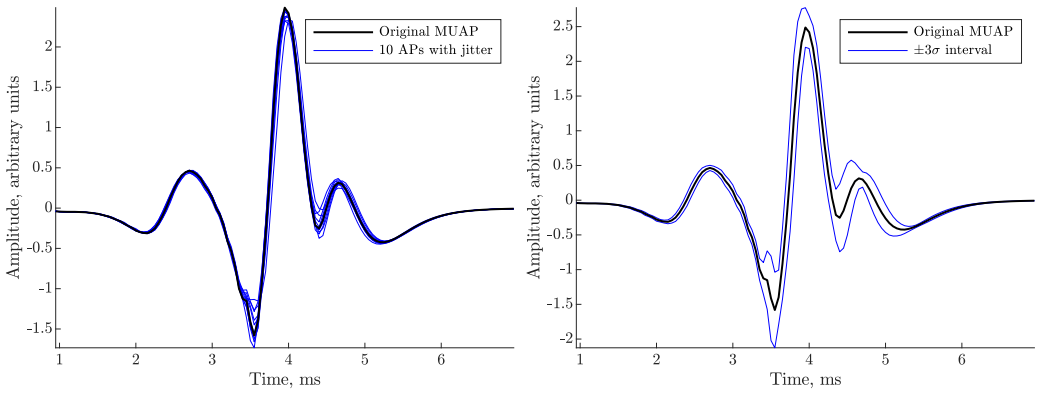
This helps to significantly accelerate the simulation since the calculation of intracellular action potentials and SFAPs for each new realization of a MUAP is no longer necessary. However, this restricts the MUAPs to be calculated only for discrete-time arguments. This disallows the simulation of time delays that are smaller than T_s, such as those caused by neuromuscular jitter. Additionally, the spike arrival times I_{nk} are, as well, not restricted to discrete-time values.

In order to resolve these issues, we calculate the total temporal delay in the argument of each SFAP in (4.15) and take the modulo δ and quotient Δ of its division by the sampling period T_s:

$$\begin{aligned} \delta &= (d_f - \zeta - I_{nk}) \bmod T_s \\ \Delta &= (d_f - \zeta - I_{nk} - \delta)/T_s \end{aligned} \tag{4.16}$$

Pre-calculated SFAPs are then simply delayed by the quotient Δ. Then, the obtained waveform $\Phi_{np}(t)$ can be shifted by delay δ using sub-sample shifting technique proposed in [30]. That is:

$$\Phi_{np}(t) = \sum_{f=1}^{F_n} \tilde{\phi}_{fp}(t - \Delta) \tag{4.17}$$



(a) (b)

Figure 4.11: Effect of the neuromuscular jitter on MUAP waveform.

where $\tilde{\phi}_{fp}(t)$ is pre-calculated vector $\phi_{fp}(t_n)$ shifted by sub-sample delay δ.

As an example, we provide an original simulated MUAP waveform (see Figure 4.11) and its multiple realizations for neuromuscular jitter with a standard deviation of 25 µs [6]. One can see that even the typical level of jitter, observed in healthy subjects, considerably affects the form of the MUAP. This underlines the importance of the possibility to simulate fine temporal delays in iEMG modelling.

4.4.4 EMG in a single observation point

EMG is modeled as a linear sum of contributions from all MUs [1], while each contribution is a convolution of a spike train and MUAP. We formulate the expression for simulated EMG, acquired in observation point p, in the following way:

$$y_p(t) = \sum_{n=1}^{N} \sum_{k=1}^{\mathrm{card}(U_n)} \Phi_{np}(t - I_{nk}) \tag{4.18}$$

where $y_p(t)$ is the simulated EMG signal in observation point p, U_n is a vector of spikes' time instants for n-th motor neuron; and k is the index of a spike in U_n.

4.4.5 EMG in a single-channel electrode

Since a metallic electrode is a conductor, the electric potential is constant across its volume. Its value can be approximated by an average field in the observation points adjacent to the electrode [6]. Therefore, a SFAP detected by an electrode can be calculated as an integral of the potential $\phi(t)$ over the electrode's surface. Due to the linearity of (4.15) and (4.18), the same applies to MUAPs and the overall signal detected by the electrode.

In our model, we approximate the recording surface by a number of elements with observation points $[p_1, p_2, \ldots, p_P]$ associated to the center of each element. The electrode potential is, thus, equal to the sum of element potentials weighted by

their areas:

$$\mathbf{Y}^{\mathbf{E}}(t) = \begin{bmatrix} \pm e_1 & \pm e_2 & \dots & \pm e_P \end{bmatrix} \cdot \begin{bmatrix} y_1(t) \\ y_2(t) \\ \dots \\ y_P(t) \end{bmatrix} = \mathbf{E}\mathbf{Y}(t) \tag{4.19}$$

where e_p is area of the electrode's element associated to p-th observation point, its sign depends on the polarity of corresponding amplifier input; $y_p(t)$ is EMG signal calculated at observation point p using equation (4.18); in following, $\mathbf{E}$ and $\mathbf{Y}(t)$ will be referred to as *electrode matrix* and *observation vector*.

As an example, a fine wire electrode can be approximated by a pair of points with equal areas. In the case of bipolar acquisition, the resulting signal is equal to the difference between potentials observed in the two points (see expression (4.19)):

$$\mathbf{Y}^{\mathbf{E}}(t) = \begin{bmatrix} 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix}$$

Similarly, single-channel MUAP can be calculated in the following way:

$$\mathbf{\Phi}_n^{\mathbf{E}}(t) = \begin{bmatrix} \pm e_1 & \pm e_2 & \dots & \pm e_P \end{bmatrix} \cdot \begin{bmatrix} \Phi_{n1}(t) \\ \Phi_{n2}(t) \\ \dots \\ \Phi_{nP}(t) \end{bmatrix} = \mathbf{E}\mathbf{Y}(t) \tag{4.20}$$

4.4.6 EMG in a multichannel electrode

Multichannel electrodes are arrays or matrices of electrodes. Calculation of a multichannel EMG signal can be conveniently represented by a stack of electrode matrices $\mathbf{E}$:

$$\mathbf{Y}^{\mathbf{E}}(t) = \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \\ \dots \\ \mathbf{E}_M \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \\ \dots \\ y_P(t) \end{bmatrix} = \mathbf{E}\mathbf{Y}(t) \tag{4.21}$$

where electrode matrix $\mathbf{E}$ has a sense of *mixing matrix*.

For example, a signal acquired by an array of point electrodes with consecutive differentiation can be represented as follows (see expression (4.21)):

$$\mathbf{Y}^{\mathbf{E}}(t) = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ & \dots & & & \\ 0 & \dots & & -1 & 1 \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \\ \dots \\ y_P(t) \end{bmatrix}$$

Multichannel MUAPs can be obtained in the similar way:

$$\mathbf{\Phi}_n^{\mathbf{E}}(t) = \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \\ \dots \\ \mathbf{E}_M \end{bmatrix} \begin{bmatrix} \Phi_{n1}(t) \\ \Phi_{n2}(t) \\ \dots \\ \Phi_{nP}(t) \end{bmatrix} = \mathbf{E}\mathbf{\Phi}_n(t) \tag{4.22}$$

4.4.7 EMG in a shifting electrode

Shifts can be modeled as a combination of translations and rotations of the electrode along a specified trajectory in the muscle. This trajectory can be approximated by a number of *nodes* D linked by successive rigid transformations $\mathbf{T}_0^1$, $\mathbf{T}_1^2$,..., $\mathbf{T}_{D-1}^D$.

The current position of the electrode on the trajectory curve can be specified by a continuous path parameter $0 \leq \lambda \leq D-1$, where D denotes the overall number of nodes in the trajectory. Observation vector $\mathbf{Y}^d(t)$ in trajectory node d is defined as follows:

$$\mathbf{Y}^d(t) = \begin{bmatrix} y_1^d(t) \\ y_2^d(t) \\ \dots \\ y_P^d(t) \end{bmatrix} \tag{4.23}$$

where $y_p^d(t)$ is observation obtained in point p^d, position of the point p after d-th transformation along the path: $p^d = \mathbf{T}_{d-1}^d \dots \mathbf{T}_0^1 \cdot p$

The signal, as a function of the current position of the electrode, can be calculated as follows:

$$\mathbf{Y}^{\mathbf{E}}(t,\lambda) = \mathbf{E}\left[\mathbf{I}_1(\lambda)\ \mathbf{I}_2(\lambda)\ \dots\ \mathbf{I}_D(\lambda)\right] \begin{bmatrix} \mathbf{Y}^1(t) \\ \mathbf{Y}^2(t) \\ \dots \\ \mathbf{Y}^D(t) \end{bmatrix} \tag{4.24}$$

where, for $d = \{1, 2, \dots D\}$, $\mathbf{I}_d(\lambda) = \mathbf{I} \cdot \delta(\lambda - (d-1))$, $\delta(\cdot)$ is the Dirac delta function and $\mathbf{I}$ is identity matrix of size P; $\mathbf{Y}^d(t)$ is the observation vector in the trajectory node d.

As an example, assuming that the electrode's trajectory is approximated by only two nodes, an EMG signal from a fine-wire electrode, before the shift ($\lambda = 0$) and after the shift ($\lambda = 1$) can be expressed as (see equation (4.24)):

$$\mathbf{Y}^{\mathbf{E}}(t,0) = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}^\top \begin{bmatrix} y_1^1(t) \\ y_2^1(t) \\ y_1^2(t) \\ y_2^2(t) \end{bmatrix}, \mathbf{Y}^{\mathbf{E}}(t,1) = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}^\top \begin{bmatrix} y_1^1(t) \\ y_2^1(t) \\ y_1^2(t) \\ y_2^2(t) \end{bmatrix}$$

where the lower index of y corresponds, as previously, to the electrode element, while its upper index denotes the trajectory node.

The signal acquired in positions located between the trajectory nodes can be linearly interpolated, given a sufficiently fine trajectory sampling. One can express the signal acquired in a specific position λ on the trajectory:

$$\mathbf{Y}^{\mathbf{E}}(t,\lambda) = \mathbf{E}\left[\hat{\mathbf{I}}_1(\lambda)\ \hat{\mathbf{I}}_2(\lambda)\ \dots\ \hat{\mathbf{I}}_D(\lambda)\right] \begin{bmatrix} \mathbf{Y}^1(t) \\ \mathbf{Y}^2(t) \\ \dots \\ \mathbf{Y}^D(t) \end{bmatrix} \tag{4.25}$$

where, for $d = \{1, 2, \dots D\}$, $\hat{\mathbf{I}}_{\mathbf{d}}(\lambda)$ is a weighted identity matrix determined as function of the electrode position λ:

$$\hat{\mathbf{I}}_d(\lambda) = \mathbf{I} \cdot \max\left[0,\ 1 - |d - 1 - \lambda|\right] \tag{4.26}$$

Function (4.26) is equal to one when $\lambda = d - 1$ (i.e. when calculating the signal exactly in trajectory node d), and linearly weights the neighboring nodes d and $d+1$ when $d - 1 < \lambda < d$. An example of a MUAP captured by a fine-wire electrode, that moved transversely to the MFs, is shown in Figure 4.12.

Applying expression (4.25) to the previous example, we can calculate the signal acquired at 1/4-th of the way ($\lambda = 0.25$):

$$\mathbf{Y}^{\mathbf{E}}(t, 0.25) = \begin{bmatrix} 0.75 \\ -0.75 \\ 0.25 \\ -0.25 \end{bmatrix}^{\top} \begin{bmatrix} y_1^1(t) \\ y_2^1(t) \\ y_1^2(t) \\ y_2^2(t) \end{bmatrix}$$

Similarly to previous examples, MUAP in a shifting electrode can be calculated as:

$$\mathbf{\Phi}^{\mathbf{E}}(t, \lambda) = \mathbf{E}\left[\hat{\mathbf{I}}_1(\lambda)\ \hat{\mathbf{I}}_2(\lambda)\ \dots\ \hat{\mathbf{I}}_D(\lambda)\right] \begin{bmatrix} \mathbf{\Phi}^1(t) \\ \mathbf{\Phi}^2(t) \\ \dots \\ \mathbf{\Phi}^D(t) \end{bmatrix} \tag{4.27}$$

Path parameter λ can be a function of force or time since usually electrode shifts occur due to either muscle deformation during contraction or other factors that can be described as functions of time.

4.4.8 Results and application example for multichannel MUAP simulation

To assess the performance of the proposed multichannel MUAP modelling, we establish a simulation case showed in Figure 4.12: an array of 16 equidistant electrodes spaced by 1 mm gap was inserted into the muscle at the angle of 30 degrees to the fibers. This array simulates recently proposed intramuscular electrodes for human studies [31]. In total, 15 channels were obtained by consecutive differentiation of the signals in the electrodes. We have chosen a MU for which the center of the territory lied close to the electrodes. The length of the muscle in this simulation was 50 mm, and the end-plate zone was positioned around the center of the muscle, as stated in formula (4.8). The array was placed in one of the halves of the muscle and didn't cross the MU's end-plate zone.

From Figure 4.12, we notice several relevant results. First, the amplitudes of the MUAPs are inversely proportional to the distance between the channel's electrodes and the center of the MU's territory. Second, the centers of energy of the MUAPs shift to the right as the distance between the channel and the end-plate increases, due to the simulation of SFAP conduction. Third, the transformations between

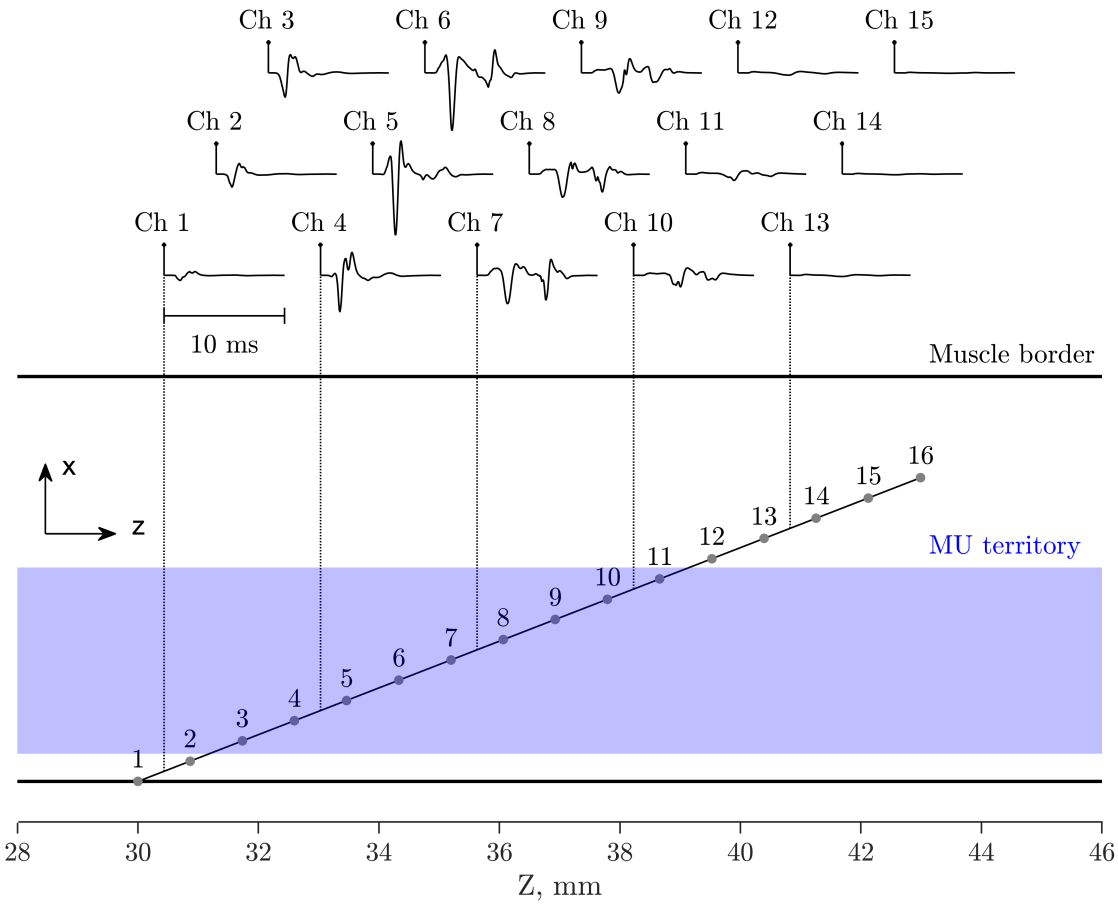


Figure 4.12: Lower half: a multichannel array of 16 equidistant electrodes spaced by 1 mm gap, inserted into the muscle at the angle of 30 degrees to the fibers. Consecutive differentiation is applied to the multichannel signal, providing 15 differential channels. Upper half: MUAPs in each of the resulting 15 differential channels. All MUAP plots have the same vertical axis scale. The black round markers designate the trigger time, similar for all the MUAP plots.

MUAPs in neighboring channels are consistent and MUAPs have physiologically correct durations (approximately 5 ms).

4.4.9 Results and application example for scanning acquisition simulation

Equation (4.25) permits to simulate a "scan" of a motor unit territory. An example is presented in Figure 4.13 where a MUAP is shown at 10 equidistant nodes positioned along a straight trajectory that goes across a large MU's territory through its center. The overall duration of the generated MUAP is approximately 5 ms. This result is qualitatively in agreement with experimental observations [9, 8].

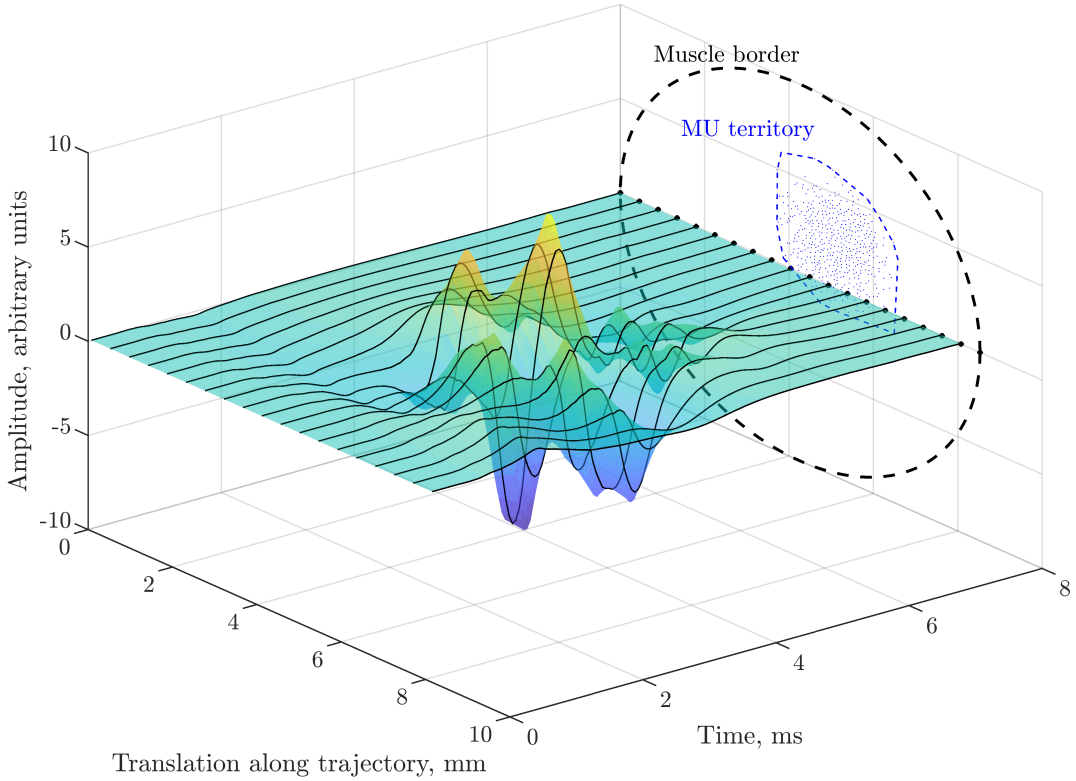
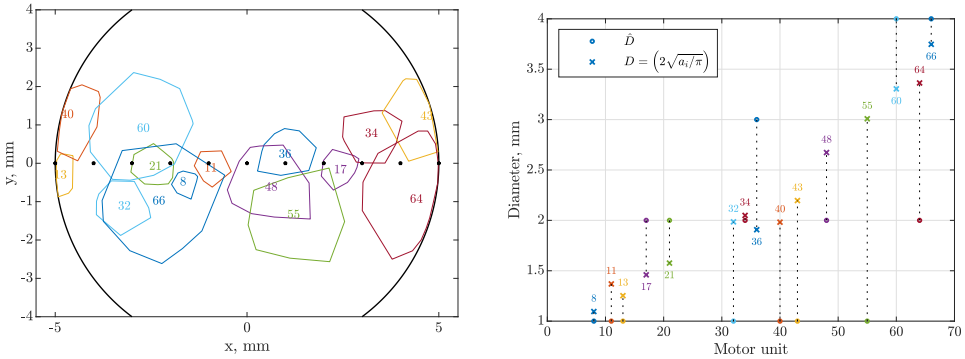


Figure 4.13: A scan simulated with a two-point differential electrode which was shifted transversely along a 10 mm-long trajectory across the territory of a MU. Black dots on the right plane are 1 mm increments of the electrode position, for which the MUAPs (black solid lines) are calculated. Intermediate values of MUAPs are obtained using scanning and interpolation formula (4.25).



(a) Location of the innervation territories of detectable MUs in the cross-section of the muscle. Black dots - electrode locations, numbers - indices of MUs, colored lines - territories' borders.

(b) Estimated ($\hat{D}$, circles) and true (D, crosses) diameters of MUs territories. Colors are matched with the left figure for easier correspondence.

Figure 4.14: Results of territory assessment simulation.

4.4.10 Results and application example for MUs territory assessment

The multichannel MUAP model permits to simulate the studies that aim at the estimation of MUs' innervation territories. As an example, here we have simulated a procedure that is similar to a previous experimental study [32]. More specifically, we have simulated a 10-mm long array of 11 equally spaced intramuscular electrodes, making ten channels with consecutive differentiation, inserted to a 10-mm wide muscle at an angle of 90° to the fibers. The diameters of the territories are estimated as the number of the channels where the MUAP's peak was four times greater than the standard deviation of the baseline noise [33, 32]. The SNR was set to 15 dB in this study. Only MUs with recruitment thresholds that are below of 50% MVC were considered for the analysis.

From Figure 4.14, the estimated diameters generally correlate with their true values. However, this simulation highlights the fact that, for most of the MUs, the electrode array does not cross the center of their innervation territories, which results in an underestimation of the diameters (e.g., see MUs 40, 43, 55 in Figure 4.14). Also, we note the low resolution of this technique which limits the estimated diameters to be multiples of the inter-electrode distance. The proposed simulation model can be used to test and evaluate more complex approaches to MU territory assessment based on multichannel iEMG decomposition.

4.5 Annotation and MUAP dictionary generation

In order to test and evaluate the decomposition and decomposition-based approaches, the simulated iEMG signal should be annotated, i.e. the underlying spike trains and MUAP dictionary should be provided. Obtaining the spike trains is straightforward: one can simply take the trains used in the simulation of the signal. However, only

those MUs that are close enough to the electrode and, thus, are detectable in the signal, should be included to the annotation. In the case of manual decomposition, this choice is made by a trained specialist, otherwise, it is done automatically by a decomposition algorithm, based on an estimate of the baseline noise.

While simulating an EMG signal, one is free to set an arbitrary level of baseline noise. This will affect the annotation of the signal, by either excluding MUs whose MUAPs are "drowned" in the noise, or adding MUs that sufficiently emerge from it. Additionally, on high levels of contraction, small MUAPs are completely masked by the larger ones even if these MUAPs were clearly detectable compared to the baseline noise. Such MUAPs, in that case, would never appear in manual or automatic decomposition due to the lack of their distinct non-superimposed occurrences in the signal. In cases when a decomposition algorithm is being evaluated, an inclusion of such MUs to the annotation may be unfair and leads to the drop of the evaluation scores.

Additionally, when establishing a simulation model for decomposition-based control, it is convenient to exclude the decomposition algorithm from the loop and use the true annotation instead. This simplifies the testing process and permits to concentrate on the evaluation of the control model and avoid decomposition-specific issues in the process.

In order to automatically produce a "fair" annotation that includes only detectable MUs, we use a two-stage approach based both on the MUAP comparison with the baseline noise and on its contribution to the overall signal. In the remainder of this section, we will present the strategy of setting the baseline noise level for simulation and to select the MUs that are prominent enough to be detected by a decomposition algorithm.

4.5.1 Instrumentation noise level setting for simulation

In experimental EMG acquisitions, there are two main sources of noise: the instrumentation and distant motor unit activity. While the former remains constant during acquisition, the latter changes depending on the contraction level.

Instrumentation noise level is usually defined in terms of Signal-to-Noise ratio (SNR), that is, as $20\log_{10}(\sigma_s/\sigma_n)$. The standard deviation of instrumentation noise can be measured in the absence of a contraction, and is considered constant. However, the standard deviation of the EMG signal grows with the contraction force due to the recruitment of larger MUs and increasing firing rates. Therefore, the SNR varies while the real level of instrumentation noise is constant.

In order to standardize the instrumentation noise level between different contraction profiles, we calculate the SNR for the EMG signal only at the maximal value of net excitation:

$$\sigma_s = 10^{-\frac{\mathrm{SNR}}{10}} * \sqrt{\frac{1}{T}\int_0^T S^2_{100\%}(t)\,\mathrm{d}t} \tag{4.28}$$

where T is sufficiently long for the contraction to stabilize at its maximum value (usually T=5 s is enough).

In multichannel EMG simulation, instrumentation noise is calculated separately for each channel and then averaged to produce a single value for all channels.

4.5.2 Selection of detectable MUs

Selection of detectable MUs for annotation is based on the prominence of their MUAPs in the overall signal. In our simulation, we select detectable MUAPs using two criteria: first is MUAP's prominence in the instrumentation noise, second is MUAP's contribution to the total signal power.

The first criterion is based on comparison of the MUAP's maximum absolute value with the standard deviation of instrumentation noise. In our simulation, a MUAP is considered detectable if its peak value is 6 times greater than σ_n.

The second criterion uses the contribution of the MUAP to the power of EMG signal. That is, we want to keep only those MUs whose APs are large enough compared to the others. We calculate the contribution of n-th MU to the signal power in the following way:

$$P_n(r_n) = \varphi_n(r_n) \int_0^T \Phi_n(t)^2 \, \mathrm{d}t \tag{4.29}$$

where $\varphi_n(\cdot)$ is the firing rate of n-th MN as function of net excitation, r_n is recruitment threshold of n-th MN and T is the duration of MUAP $\Phi_n(t)$. Relative contribution of n-th MU is:

$$C_n = \frac{P_n(r_n)}{\sum_{i=1}^{N} P_i(r_n)} \tag{4.30}$$

Thus, the contribution is calculated at the level of contraction corresponding to the recruitment threshold of the MN. That is, we state that if the MUAP is not prominent at the recruitment level of its MN, it will not become prominent at a higher level of contraction. A MU is considered detectable if its relative contribution is greater than a specified threshold. In our simulations we use the rule $C_n \geq 0.01$. This criterion excludes those MUs from the annotation, whose MUAPs will be repeatedly masked by the larger ones and thus won't be detectable.

Finally, a MU is added to the annotation if its MUAP satisfies both criteria: its peak is greater than six times the noise standard deviation, and its relative contribution is higher than a specified threshold. In the cases of multichannel iEMG simulation, a MU is considered detectable if the rules are satisfied in at least one channel. This approach produces choices that are close to the ones made intuitively by an expert or a by detection phase of a decomposition algorithm.

4.5.3 Annotation and dictionary generation for a simulated signal

As we have stated earlier, decomposition annotation consists of simulated spike trains of detectable motor units. In our implementation, the spike timings coincide with the moments of MNAP arrival to the first branching point of the terminal

arborization. Due to the integration of MNAP and SFAP propagation times, the resulting MUAPs appear with a specific delay relative to the spike timings.

In order to provide an annotation that is consistent with the corresponding iEMG signal, we need to compensate for these delays. We estimate the delay for each MU and shift the corresponding spike train to align the spikes with the centers of the MUAPs. In order to estimate the position of the center of the MUAP of n-th MU we identify the time τ_n for which:

$$\int_{-\infty}^{\tau_n} \Phi_n(t)^2 \, \mathrm{d}t = \int_{\tau_n}^{\infty} \Phi_n(t)^2 \, \mathrm{d}t \tag{4.31}$$

In case of a multichannel acquisition, for each MU, the delay is calculated using channel c where its MUAP has maximal energy:

$$\int_{-\infty}^{\infty} \mathbf{E_c} \Phi_n(t)^2 \, \mathrm{d}t \tag{4.32}$$

In the case of the electrodes shifting to distances longer than 1 mm, the delays become variable and difficult to track. We did not implement any specific approach to this problem. It can be solved by recalculating the delays in each node of the electrode trajectory using (4.31) and (4.27).

Dictionary of MUAPs can be generated using formula (4.27) for each channel and in each node of the trajectory. Additionally, MUAPs are centered and their lengths should be equalized. We set standard length of a MUAP to 5 ms for simulations that don't include contractions exceeding 50% MVC. For other cases, larger MUs are recruited, so that the dictionary should be extended to 7.5 ms or 10 ms to fit the larger MUAPs.

Finally, an annotation that positions the spike at the center of the MUAP is considered non-causal since the first half of a MUAP appears before the corresponding spike. Such annotation is used in EMGLAB [34]. In order to produce a causal annotation, spikes should be shifted to the left by the half of the MUAP length. This type of annotation is of use when testing our Bayesian-filtering based decomposition algorithm [35].

4.5.4 Results and application example for multichannel iEMG decomposition

We present the results of the application of a decomposition algorithm to the simulated signals. In order to generate an iEMG signal for decomposition, we have implemented the motor neuron pool model proposed in [7].

A linear array of five electrodes (1-mm interelectrode distance) with consequent differentiation, providing four iEMG channels, was used for this simulation. We measured the average power of the simulated signal in all channels at maximal net excitation in order to obtain a reference value for calculation of the standard deviation σ of the additive noise. MUAPs whose maximal absolute value exceeded 4σ in at least one of the four channels were considered detectable and their MUs

were included to the annotation of the signal. The SNR in all channels was set to 15dB and a trapezoidal contraction reaching 20% MVC was generated.

The simulated EMG signal was decomposed by MTL, the multichannel version of the algorithm proposed in [36, 34]. We have compared the simulated annotation with the decomposition provided by MTL. The decomposition was evaluated using classification phase sensitivity and positive predictivity [37] averaged across 11 detected MUs, which resulted in, respectively, 0.93 ± 0.04 and 0.97 ± 0.03.

4.6 Conclusion

Simulation signal is a key tool for understanding the nature of intramuscular EMG. Signals simulated in a fully controlled environment help to test and validate various approaches to iEMG processing. In this chapter, we have presented a set of models that, together, simulate physiologically correct MUAPs and iEMG signals.

More specifically, we combine existing methods with novel approaches in order to achieve such goals as:

- Controllable and stable simulation of MU territories;
- Ability to simulate multichannel recordings for arbitrary and time-varying locations of the electrode in the muscle;
- Ability to constrain the complexity of MUAPs waveforms and establish their consistency in multichannel recordings.

The first goal is achieved using the farthest point sampling algorithm applied to the generation of MU territory centers, and of Gaussian-based modelling of muscle fibers innervation. The second is addressed by modelling the electrode as a set of points having equal potential in the electric field generated by surrounding muscle fibers. Multichannel and scanning recordings are obtained using linear transformations of the signals calculated in such points, taking into consideration the polarity. The third goal is achieved by means of multi-modal distribution of the neuromuscular junctions along the muscle fibers, which defines the action potential propagation delays and directly affects the MUAP shapes.

Finally, we have presented examples of applications of the simulated signals, such as MU territory assessment, iEMG decomposition, multichannel MUAP simulation. Using this model along with the one presented in the following chapter, we will establish a prosthetic control simulation model for tests and validation of intent estimation techniques. The model was implemented on Matlab and is accessible on-line on the authors' repository [1].

[1]Link: https://github.com/smetanadvorak/iemg_simulator

Bibliography

[1] D. Farina, A. Crosetti, and R. Merletti, "A model for the generation of synthetic intramuscular EMG signals to test decomposition algorithms," *IEEE transactions on biomedical engineering*, vol. 48, no. 1, pp. 66–77, 2001.

[2] J. L. Dideriksen, D. Farina, and R. M. Enoka, "Influence of fatigue on the simulated relation between the amplitude of the surface electromyogram and muscle force," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 368, pp. 2765–2781, June 2010.

[3] G. Venugopal, P. Deepak, D. M. Ghosh, and S. Ramakrishnan, "Generation of synthetic surface electromyography signals under fatigue conditions for varying force inputs using feedback control algorithm," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 231, pp. 1025–1033, Nov. 2017.

[4] S. D. Nandedkar and E. Stalberg, "Simulation of single muscle fibre action potentials," *Medical & Biological Engineering & Computing*, vol. 21, pp. 158–165, Mar. 1983.

[5] A. Miller-Larsson, "Motor unit action potential field ? Modelling results," *Biological Cybernetics*, vol. 53, pp. 307–321, Mar. 1986.

[6] A. Hamilton-Wright and D. Stashuk, "Physiologically Based Simulation of Clinical EMG Signals," *IEEE Transactions on Biomedical Engineering*, vol. 52, pp. 171–183, Feb. 2005.

[7] A. J. Fuglevand, D. A. Winter, and A. E. Patla, "Models of recruitment and rate coding organization in motor-unit pools," *Journal of neurophysiology*, vol. 70, no. 6, pp. 2470–2488, 1993.

[8] J. Navallas and E. Stalberg, "Studying motor end-plate topography by means of scanning-electromyography," *Clinical Neurophysiology*, vol. 120, pp. 1335–1341, July 2009.

[9] E. Stalberg and L. Antoni, "Electrophysiological cross section of the motor unit.," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 43, pp. 469–474, June 1980.

[10] D. Farina, "A Novel Approach for Precise Simulation of the EMG Signal Detected by Surface Electrodes," *IEEE Transactions on biomedical engineering*, vol. 48, no. 6, pp. 637–646, 2001.

[11] E. Petersen and P. Rostalski, "A Comprehensive Mathematical Model of Motor Unit Pool Organization, Surface Electromyography, and Force Generation," *Frontiers in Physiology*, vol. 10, p. 176, Mar. 2019.

[12] E. Henneman, G. Somjen, and D. O. Carpenter, "Functional significance of cell size in spinal motorneurons," *Journal of Neurophysiology*, vol. 28, pp. 560–580, May 1965.

[13] D. Farina and A. Rainoldi, "Compensation of the effect of sub-cutaneous tissue layers on surface EMG: A simulation study," *Medical Engineering & Physics*, vol. 21, pp. 487–497, July 1999.

[14] V. Carriou, J. Laforet, S. Boudaoud, and M. A. Harrach, "Realistic motor unit placement in a cylindrical HD-sEMG generation model," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, (Orlando, FL, USA), pp. 1704–1707, IEEE, Aug. 2016.

[15] G. Peyré, "Numerical Mesh Processing. Course notes.," p. 75, 2008.

[16] Y. Eldar, M. Lindenbaum, and M. Porat, "The Farthest Point Strategy for Progressive Image Sampling," *IEEE Transactions on Image Processing*, vol. 6, no. 9, pp. 1305–1315, 1997.

[17] D. P. Mitchell, "Generating antialiased images at low sampling densities," in *In SIGGRAPH '87: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, ACM*, Press, 1987.

[18] M. A. Johnson, J. Polgar, D. Weightman, and D. Appleton, "Data on the distribution of fibre types in thirty-six human muscles: An autopsy study," *Journal of the Neurological Sciences*, vol. 18, pp. 111–129, Jan. 1973.

[19] J. Navallas, A. Malanda, L. Gila, J. Rodriguez, and I. Rodriguez, "Comparative evaluation of motor unit architecture models," *Medical & Biological Engineering & Computing*, vol. 47, pp. 1131–1142, Nov. 2009.

[20] J. Navallas, A. Malanda, L. Gila, J. Rodríguez, and I. Rodríguez, "A muscle architecture model offering control over motor unit fiber density distributions," *Medical & Biological Engineering & Computing*, vol. 48, pp. 875–886, Sept. 2010.

[21] M. H. Brooke and W. K. Engel, "The histographic analysis of human muscle biopsies with regard to fiber types. 1. Adult male and female," *Neurology*, vol. 19, pp. 221–233, Mar. 1969.

[22] E. Petersen and P. Rostalski, "A Comprehensive Mathematical Model of Surface Electromyography and Force Generation," *Cold Spring Harbor Laboratory*, p. 32, Feb. 2018.

[23] F. Buchthal, C. Guld, and P. Rosenfalck, "Multielectrode Study of the Territory of a Motor Unit," *Acta Physiologica Scandinavica*, vol. 39, pp. 83–104, Jan. 1957.

[24] R. E. Burke, "Motor Units: Anatomy, Physiology, and Functional Organization," in *Comprehensive Physiology* (R. Terjung, ed.), pp. 345–422, Hoboken, NJ, USA: John Wiley & Sons, Inc., Jan. 2011.

[25] D. A. Gibson and L. Ma, "Developmental regulation of axon branching in the vertebrate nervous system," *Development*, vol. 138, pp. 183–195, Jan. 2011.

[26] P. Rosenfalck, "Intra- and extracellular potential fields of active nerve and muscle fibres. A physico-mathematical analysis of different models," *Acta Physiologica Scandinavica. Supplementum*, vol. 321, pp. 1–168, 1969.

[27] R. Plonsey, "The Active Fiber in a Volume Conductor," *IEEE Transactions on Biomedical Engineering*, pp. 371–381, 1974.

[28] J. Malmivuo and R. Plonsey, *Bioelectromagnetism Principles and Applications of Bioelectric and Biomagnetic Fields*. Oxford University Press, Oct. 1995.

[29] E. Stålberg and J. V. Trontelj, "The study of normal and abnormal neuromuscular transmission with single fibre electromyography," *Journal of Neuroscience Methods*, vol. 74, pp. 145–154, June 1997.

[30] K. C. McGill and L. J. Dorfman, "High-Resolution Alignment of Sampled Waveforms," *IEEE Transactions on Biomedical Engineering*, vol. 31, pp. 462–468, June 1984.

[31] S. Muceli, W. Poppendieck, F. Negro, K. Yoshida, K. P. Hoffmann, J. E. Butler, S. C. Gandevia, and D. Farina, "Accurate and representative decoding of the neural drive to muscles in humans with multi-channel intramuscular thin-film electrodes: Multi-channel intramuscular EMG electrode," *The Journal of Physiology*, vol. 593, pp. 3789–3804, Sept. 2015.

[32] B. L. Luu, S. Muceli, J. P. Saboisky, D. Farina, M. E. Héroux, L. E. Bilston, S. C. Gandevia, and J. E. Butler, "Motor unit territories in human genioglossus estimated with multichannel intramuscular electrodes," *Journal of Applied Physiology*, vol. 124, pp. 664–671, Mar. 2018.

[33] M. E. Héroux, H. J. Brown, J. T. Inglis, G. P. Siegmund, and J.-S. Blouin, "Motor units in the human medial gastrocnemius muscle are not spatially localized or functionally grouped," *The Journal of Physiology*, vol. 593, pp. 3711–3726, Aug. 2015.

[34] K. C. McGill, Z. C. Lateva, and H. R. Marateb, "EMGLAB: An interactive EMG decomposition program," *Journal of Neuroscience Methods*, vol. 149, pp. 121–133, Dec. 2005.

[35] T. Yu, K. Akhmadeev, E. L. Carpentier, Y. Aoustin, R. Gross, Y. Pereon, and D. Farina, "Recursive decomposition of electromyographic signals with a varying number of active sources: Bayesian modelling and filtering," *IEEE Transactions on Biomedical Engineering*, pp. 1–1, 2019.

[36] J. Florestal, P. Mathieu, and A. Malanda, "Automated decomposition of intramuscular electromyographic signals," *IEEE Transactions on Biomedical Engineering*, vol. 53, pp. 832–839, May 2006.

[37] D. Farina, R. Colombo, R. Merletti, and H. Baare Olsen, "Evaluation of intramuscular EMG signal decomposition algorithms," *Journal of Electromyography and Kinesiology*, vol. 11, pp. 175–187, June 2001.

Chapter 5

Simulation of muscle contraction

5.1 Introduction

The model presented in the previous chapter permits to simulate motor unit action potentials (MUAPs). In order to generate EMG, they should be convolved with individual spike trains of the corresponding motor neurons (MNs). Hence, the presented model should be further extended by spike generation model.

Originally, spikes, or *firings*, are produced by motor neurons (MNs) of the spinal cord during the polarization-depolarization process. The last one is caused and regulated by synaptic inputs to the motor neurons, consisting of supraspinal command and internal spinal-level feedbacks. Following the approach used in other studies [1], we will consider a single synaptic input, common to all the motor neurons, termed *common drive* [2] or, as we will further refer to it, the *excitation*.

Contraction force is an increasing function of excitation. That is, larger values of excitation entail larger numbers of recruited MNs and more intensive firing activity of already recruited ones [2]. The firing behavior of MNs is usually described in terms of instantaneous frequency of their firings, or *firing rate* (FR). As we will describe in the next section, the relationship between the excitation and firing rate, called *excitation-rate curve* is individual for each MN and depends on its size. Study of this exact relationship requires a complex experimental setup [3]. More often, a *force-rate* curve is studied [4], which also exhibits an onion-skin relationship.

Firing instants of an individual MN can be modeled as a binary stochastic process with intervals between the spikes modulated by its FR. Most of the existing approaches [5, 6] model the inter-spike interval (IPI) as normally distributed with mean inversely proportional to the FR. IPIs are then added-up the in order to define the firing timings. Another approach [7] is to use survival models and Markov chains in order to generate firings in a sample-by-sample recursive fashion.

Spike trains simulated using a combination of the aforementioned methods can then be used to generate EMG signals (knowing the MUAP shapes) and, having an appropriate model of the muscle, to simulate a contraction force level. Such a model builds up similarly to the linear EMG model. That is, short elementary contractions of MUs provoked by singular spikes, called *twitches* [8], are summed up to produce total contraction power of the muscle. This approach is widely used in

other contraction force simulation models [9, 6].

The final challenge of EMG modelling comes from the fact that typical target contraction is defined in the terms of kinematic parameters of its trajectory or, more often, contraction force. That is, to generate target force, an appropriate excitation should be applied to the model of firings instants. However, it is not possible to obtain a closed-form inverse of the firing and force models. Following existing approaches [10, 11, 12], we use feedback controller and nonlinearity compensation in order to resolve this issue.

The rest of this chapter is organized as follows. First, a model of the excitation-rate curve will be introduced, followed by the description of firing instants generation approach. Next, twitch and total contraction force models will be provided. Then, static nonlinearity compensation, polynomial model of the excitation-force relationship and PID controller tuning will be presented. Finally, we will cover some implementation details of both the EMG and contraction force models.

5.2 Contraction force simulation model

5.2.1 Excitation-rate curves

Firing rate of a motor neuron is a function of its size and excitation. In [5], this relationship is modeled as linear, following experimental observations [3]. In our simulation we use the following linear model proposed in [5]:

$$\varphi_i(\varepsilon) = \begin{cases} \min\left(\varphi_i^{\max},\ \varphi_i^{\mathrm{sl}}(\varepsilon - r_i) + \varphi_i^{\min}\right), & \varepsilon > r_i \\ 0, & \varepsilon \le r_i \end{cases} \tag{5.1}$$

where ε is excitation to the motor neuron pool ($0 \le \varepsilon \le 1$); $\varphi_i^{\min}$ and φ_i^{max} are the minimum and maximum firing rates of i-th MN; r_i is its recruitment threshold and φ_i^{sl} is the slope of its rate-excitation relationship. The form of this relationship is illustrated in Figure 5.2.1.

The values of $\varphi_i^{\min}$, φ_i^{max} and g_i are themselves linear functions of recruitment rates, according to previous simulation studies [5, 13]. Parameters of these relationships will be provided in section 6.1.

5.2.2 Firing instants modelling

In order to generate the firing instants of motor neurons, most of the existing methods [5, 6] use the following approach. That is, k-th firing instant for i-th MN is calculated as follows:

$$I_{i,k} = I_{i,k-1} + \Delta_{i,k}(\varepsilon) \tag{5.2}$$

where $\Delta_i(\varepsilon)$ is i-th MN's next inter-spike interval (IPI), a random value with distribution usually modeled as normal one $g\big(\mu_i(\varepsilon), c_v(\varepsilon)\mu_i(\varepsilon)\big)$; where $\mu_i(\varepsilon) = 1/\varphi_i(\varepsilon)$ and c_v is a coefficient of variation usually set to a constant value between 1/10 and 1/3 [5].

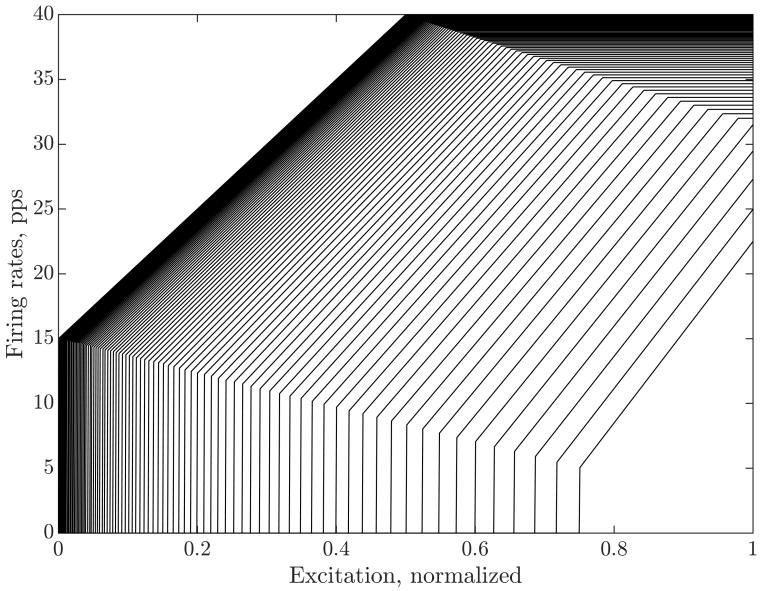


Figure 5.1: Excitation-rate relationships of motor neurons modeled as linear saturated functions, the rate is expressed in pulses per second (pps). Increasing space between the lines is the result of exponential model of recruitment thresholds ([5], also see Section 4.2.1).

This approach permits to modulate both the IPI and its variation in function of excitation. However, it uses *current* value of excitation in order to produce *future* spike timing, disregarding possible changes of excitation between the two. This is not in agreement with more precise leaky fire-and-integrate models of neurons, in which the inter-spike intervals depend on the entirety of excitation values between the two spikes. This can be critical at low firing rates of up to 10 Hz when the spiking model is unresponsive to the changes of excitation during at least 100 ms.

This problem can be addressed by using the following probabilistic model. We will use hazard function to define the probability of MN firing at each time step. Let us denote the PDF of IPI as $\Pr(\Delta_i = t) = V(t, \Theta_i(\varepsilon))$, a unimodal distribution with parameters $\Theta_i(\varepsilon)$, functions of excitation. If this PDF has a closed-form hazard function:

$$h(t, \Theta_i) = \Pr(\Delta_i = t | \Delta_i \geq t, \Theta_i) = \frac{V(t, \Theta_i)}{1 - \int_0^t V(\bar{t}, \Theta_i)\, \mathrm{d}\bar{t}} \tag{5.3}$$

then the firing instants can be generated stochastically using the Markov chain model proposed in [7]. That is, similarly to Section 3.3.1, we define $T_i[t]$ as sojourn time or time passed since the previous spike of $i-th$ MN at time instant t. As an example, $T_i[t] = 0$ signifies that i-th MN has fired at time n. Markov assumption says that the probability distribution of the next state of MN depends only on its current state. That is:

$$\Pr\left(T_i(t+1)=\bar{t}|T_i(t)\right)=\begin{cases} h\left(T_i(t)+1,\Theta_i(t)\right) & \text{if } \bar{t}=0 \\ 1-h\left(T_i(t)+1,\Theta_i(t)\right) & \text{if } \bar{t}=T_i(t)+1 \\ 0 & \text{otherwise} \end{cases} \tag{5.4}$$

This expression permits to generate spike trains in a recursive sample-by-sample way. In order to modulate this process in function of excitation, the parameter that defines the position of the PDF mode (i.e., the most probable value of IPI), usually referred to as *scale* parameter, should be set to the inverse of the firing rate. Variation of the IPI is usually modeled using coefficient of variation $c_v = \sigma^2/\mu$, which is a parameter specific to normal distribution. When using other distributions to model IPI, the variation is set by the *scale* parameter. Its appropriate value can be obtained by finding the best fit of that PDF to the normal one available from the experiments. We note here that excitation-dependent parameters $\Theta[t]$ in (5.4) affect the transition probability at each time instant, thus fast changes of excitation can be taken into account.

Possible choice of IPI PDF is, e.g., Discrete Weibull distribution [7], which has a closed-form hazard function and easily interpretable location parameter. However, its scale parameter is hard to tune when trying to generate IPIs with the coefficient of variation close to that available in the literature for normal distribution. For that reason, logistic PDF can be used instead (see Section 3.3.1). It has its hazard function in closed form, and can approximate normal distribution by setting $\mu_l = \mu_n$ and $\sigma_l = \sigma_n/1.702$ [14], where μ_l, σ_l are location and scale parameters of logistic PDF, μ_n, σ_n are the mean and standard deviation of normal PDF.

We note that this approach, having the aforementioned advantage, is more computationally demanding. That is, the hazard function should be calculated at each time instant, making the complexity of our approach proportional to the sampling frequency. In the previous approach, expression (5.2) is evaluated only at each spike timing. However, this problem can be effectively solved by generating spikes at lower rates than the rest of the simulation.

5.2.3 Twitch model

The ensemble of all muscle fibers (MFs) belonging to one single motor unit (MU) is called *muscle unit*. The arrival of a motor neuron action potential (MNAP) to the neuromuscular junctions (NMJs) of a muscle unit provokes short contraction of all its MFs. The average time duration of these elementary contractions (30ms - 100ms [15]) is large compared to the variance of delays of MNAP propagation in the terminal arborization. This permits to neglect the modeling of the individual fiber contractions and model the contraction of a muscle unit as a whole. That is, the combined contraction of an entire muscle unit is usually referred to as *twitch*.

At high firing rates, a train of subsequent spikes drives the muscle unit to create a continuous (*tetanic*) contraction that consists of overlapped (fused) twitches. At low firing rates, smoothness of the global muscle contraction is achieved by recruiting multiple motor units, all exhibiting an asynchronous firing behavior. Several twitch

models are proposed in the literature [5, 6], all built around the critically damped second-order linear system impulse response:

$$f_i(t) = \frac{P_i}{T_i} \cdot t \cdot \exp(-t/T_i) \tag{5.5}$$

where P_i and T_i are i-th motor unit's peak twitch force and time until reaching the peak. These parameters depend on the size of the corresponding MN. In our simulation, for their values, we use the assignment procedure proposed in [5].

5.2.4 Total force produced by a muscle

This section demonstrates the classic model of the contraction force [5], we provide it here for consistency of the simulation model description. Total muscle force can be modeled by a linear sum of contributions from all motor units:

$$F(t) = \sum_{i=1}^{N} \sum_{k=1}^{|I_i|} f_i(t - I_{i,k}) \tag{5.6}$$

where index i denotes i-th motor unit, k denotes k-th spike in its spike train; $f_i(t)$ is i-th motor unit's twitch function (5.5), I_i is vector of i-th motor unit's firing instants.

Force produced by a single motor unit exhibits a nonlinear behavior in function of the *stimulus rate* (a value inversely proportional to current inter-spike interval Δ) [5]. In general, the higher the stimulus rate is, the more the produced force deviates from its linearly determined value towards higher ones. This phenomenon is usually referred to as the change of *nerve-muscle system gain*. We use a model of gain proposed in [5], where at low firing rates, when $T_i/\Delta_k \leq 0.4$, gain g_{ik} is assigned to 1, otherwise:

$$g_{ik} = \frac{1 - \exp(-2(T_i/\Delta_k)^3)}{T_i/\Delta_k}, \quad T_i/\Delta_k > 0.4 \tag{5.7}$$

so that gain is a sigmoid function of excitation with peak at $T = \Delta$. Final equation for total muscle contraction force is, thus [5]:

$$F(t) = \sum_{i=1}^{N} \sum_{k=1}^{|I_i|} g_{i,k} \cdot f_i(t - I_{i,k}) \tag{5.8}$$

That is, total contraction force is a sum of spike trains convoluted with corresponding twitch waveforms and weighted by a non-linear rate-depended coefficient.

5.3 Target profile

The final goal of this simulation and the one presented in the previous chapter (see Chapter 4) is to establish a complete model that generates both the contraction that

follows a specified profile and an accompanying iEMG signal. However, the already presented parts of the model are not yet sufficient for the generation of specific contraction profiles.

While the input of the motor neuron pool is represented by excitation, target profiles are usually expressed in terms of contraction force or joint angle. Thus, in order to make the simulated muscle follow a target profile, one should either invert the muscle's excitation-force behavior or implement a closed-loop controller [1, 10]. In our simulation, we combine both approaches [6].

Excitation-force nonlinearity compensation and closed-loop controller roughly approximate the joint contributions of the supraspinal command and internal feedbacks appearing on the spinal level, such as Renshaw cells, muscle spindle, and Golgi tendon outputs. Figure 5.2 illustrates this principle. We also focus solely on the force generation and do not establish any biomechanical model that would transform this force into the mechanical effect, such as joint angle or velocity.

5.3.1 Compensation of excitation-force nonlinearity

The motor command pathway and bio-mechanical system make the relation between initial command (issued by motor cortex) and its kinematic manifestation very complex. However, humans are able to predict the kinematics of their body with high precision, inferring on its current state and currently applied motor command. This signifies that our motor cortex should have some model of this relationship in order to efficiently compensate for it.

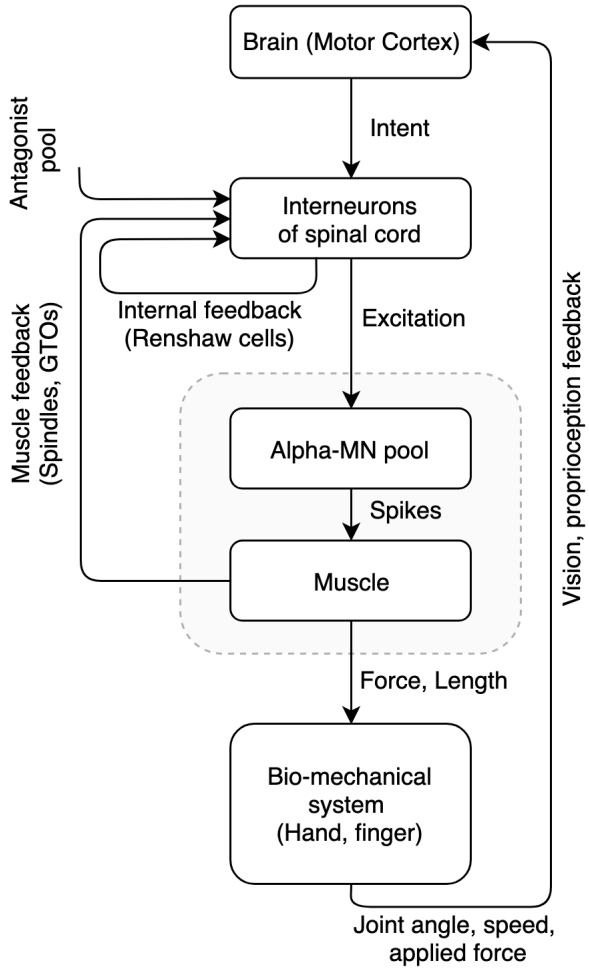
We would like to introduce this idea into our model by adding a compensation block that will directly transform the force intent into the excitation needed to achieve it. Spike trains in equation (5.8) are functions of firing rate, which, in their turn, are functions of excitation. A closed mathematical form of the inverse of this relationship (force-excitation) is not achievable, thus we are going to approximate it instead.

We generate a slow ramp contraction with the excitation increasing linearly from 0 to 1 during a very long period of time (60 seconds). Then, we calculate the resulting force using (5.8) and fit a 5-order polynomial curve to the relationship between force and excitation. We weight the measures inversely proportionally to the excitation values in order to obtain a better fit in the low values of force (see Figure 5.3).

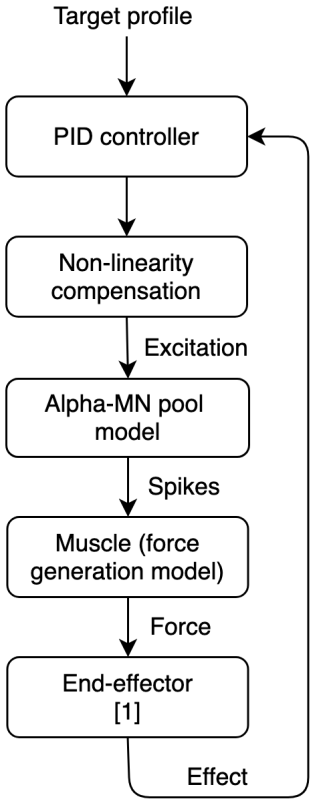
5.3.2 Identification of excitation-force model

To compensate for inconsistencies of the previously defined force-excitation model during dynamic movements, we will introduce a closed-loop regulator of the contraction force. As shown in Figure 5.2, it receives the target profile as input while its output acts on the rest of the system.

In order to tune a PID controller, we approximate the rest of the system with an output-error polynomial model. That is, the firing model, the force model, and the nonlinearity compensation are replaced by a single block (see Figure 5.4).



(a) Schematic representation of motor control system.



(b) Schematic representation of simulated motor control system.

Figure 5.2: (a) Schematic representation of motor control system. Dashed line encloses the parts for which we have established a simulation model. (b) Schematic representation of simulated motor control system. The target profile replaces the intent; the end-effector block's transfer function is 1, meaning that we do not model the bio-mechanical part of the system and consider the contraction force to be the target output.

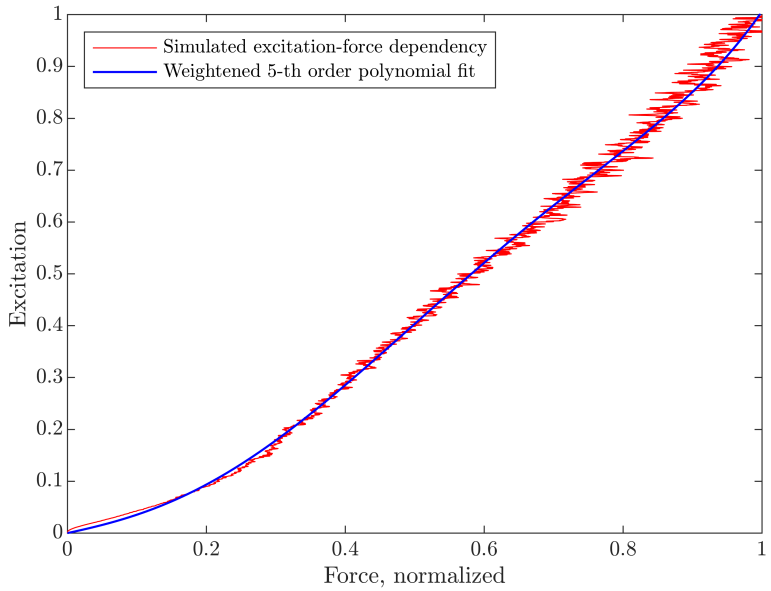
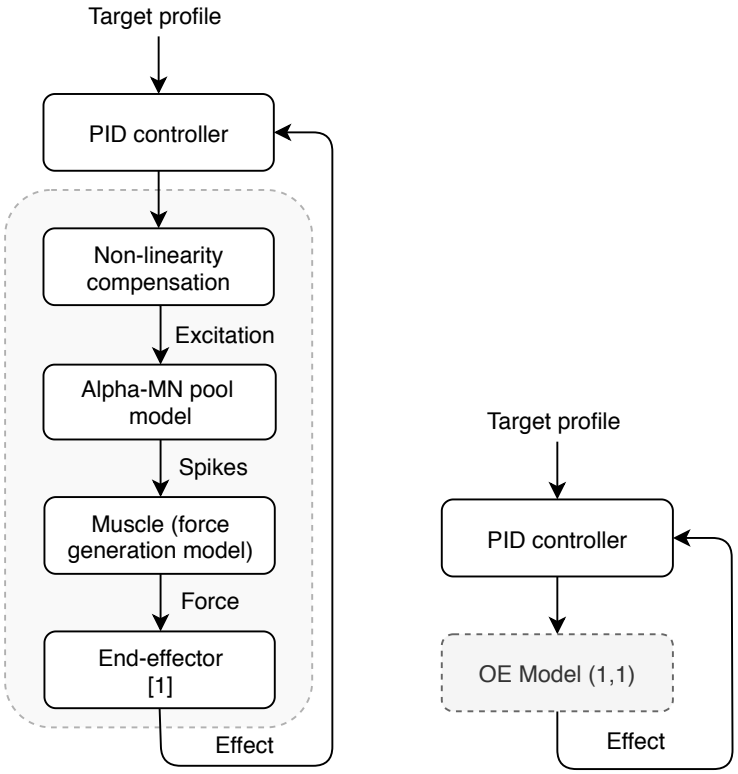


Figure 5.3: Red: Excitation-force data from a simulated slow contraction with excitation linearly increasing from 0 to 1. Force is normalized to its maximum value obtained by averaging force measures at excitation equal to 1. Blue: Fifth-order polynomial fit obtained by weighted least squares. Weights are assigned to each sample, depending on its excitation value: $w(\varepsilon) = (1.25 - \varepsilon)/1.25$, so that $w \in [0.2, 1]$. The fitting curve is forced to pass through 0.



(a) (b)

Figure 5.4: Blocks transforming the excitation to force (enclosed by gray area on (a)) are altogether replaced by a polynomial first order output-error model to tune the PID controller (b).

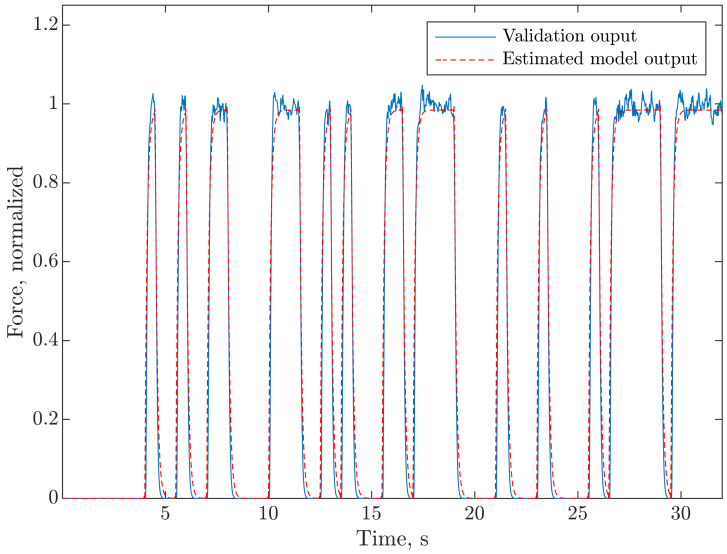


Figure 5.5: Identification and validation of the linear OE model of excitation-force relation dynamic.

In [16], the excitation-force model is constructed of two first-order low-pass filters representing pool activation and muscle contraction dynamics. For simplicity, we don't establish separate models for these processes and instead we consider them one joint first-order low-pass filter.

We estimate its parameters from muscle's response to a step excitation, using the least-squares method (Matlab's *oe* function). The identification and validation signals consisted of binary rectangular excitation pulses of random duration (0.5 - 5 seconds). The validation results are shown in Figure 5.5, obtained output-error model is $F(z) = 1 - 0.82z^{-1}$ at sampling rate of 50 Hz. We consider the obtained validation results satisfying. The first-order low-pass filter closely approximates the overall muscle dynamics during zero-to-maximal contractions.

5.3.3 Tuning of a PI controller

To calculate the necessary excitation at the input of the motor neuron pool, we introduce a PI controller. An example of such an approach can be found in [10]. The P and I parameters are tuned using *pidtune* function from Matlab's Dynamic System toolbox. We didn't impose any specific requirements on response time/robustness trade-off, using default parameters of the *pidtune* function.

After the coefficients were identified, we have manually adjusted the output of the controller. First, it is artificially bounded into $[0, 1]$ interval. Second, the integrated error is reset to zero each time the target force reaches zero to avoid trailing output that cannot be compensated since excitation cannot take negative values and there is no antagonist muscle in the model. Results for trapezoidal contraction reaching 100% MVC are shown on figure 5.6. PI coefficients for this example are $K_p = 0.2$,

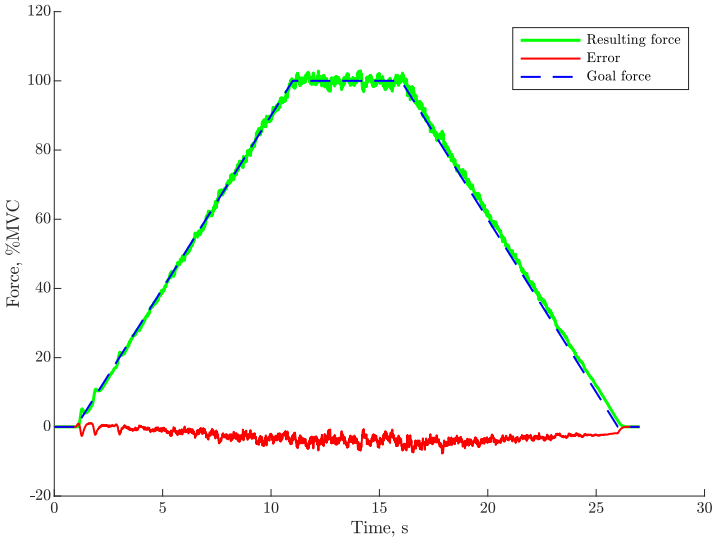


Figure 5.6: An example of simulated contraction with trapezoidal target profile.

$K_i = 10.4$ at sampling rate of 50 Hz. Simulation parameters for this model are listed in Section 6.1.

Resulting simulated force closely follows the target profile, at the same time exhibiting oscillations. This is also observed in the experimental data when the subject fails to exactly follow the profile, which results in a series of errors and over-compensations. PID controller, together with the nonlinearity compensation, permits to simulate muscle contractions with a specific target profile. This will be used in the next chapter (see Chapter 6) to generate simulated force-decomposition data-sets for training and evaluation of an effect estimation model.

5.4 Implementation details

The project with Matlab code of this simulation model is available on the authors' repository [1]. In this section, we would like to highlight some details of its implementation.

5.4.1 Global simulation model structure

The presented force simulation model works jointly with the iEMG simulation model described in the previous chapter. That is, the common part of both is the motor neuron pool, as well as the feedback controller and the nonlinearity compensation block (see Figure 5.7). Motor neuron pool produces spikes in function of the excitatory input provided by the feedback controller, according to excitation-rate curves (5.1) and spike generation models (5.2) or (5.4). Simulated spikes are then passed to both the force generation model (5.8) and the EMG model (4.18). The output of

[1] Link: https://github.com/smetanadvorak/iemg_simulator

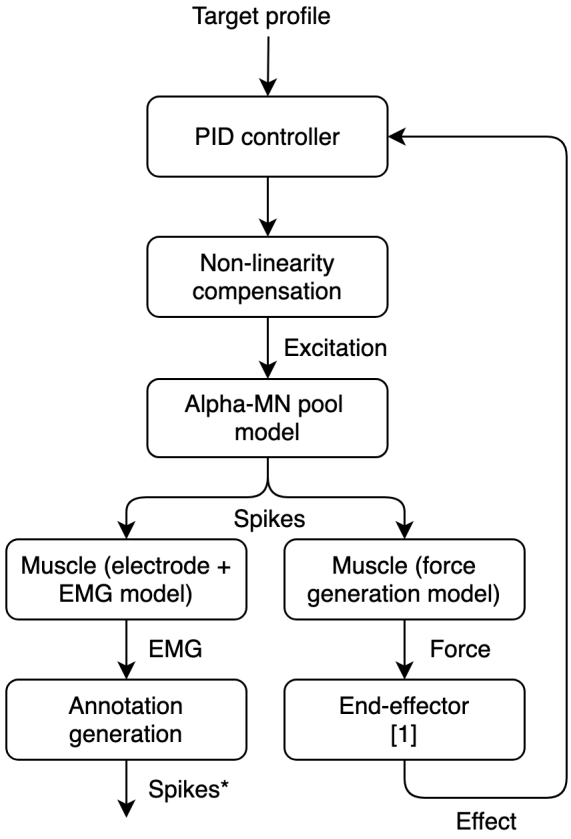


Figure 5.7: Global outline of the simulation model.

the force generation model is potentially passed to a biomechanical model that converts a single force or multiple forces into a mechanical effect. We do not attempt to establish such a model and replace it with a unity transfer function. The resulting effect is then passed to the feedback controller to adjust the excitation and follow the target profile.

5.4.2 Class structure

In order to simplify the work with the model we have implemented several Matlab classes:

- *Motor neuron pool* class that encapsulates the MN sizes, recruitment thresholds, innervation centers, and areas, as well as functions that generate them;
- *Motor unit pool* class encapsulates an instance of the previous one, the coordinates of muscle fibers and their assignment to the MNs, the lengths of axon branches, and delays of MNAP propagation;
- *Electrode* class permits to generate different types of electrodes and specify

their position and orientation in the muscle. It outputs the observation points and their positions in all trajectory nodes for SFAPs calculation;

- *Motor unit* class uses *Electrode* and *Motor unit pool* objects as arguments and encapsulates the functions that permit to generate of SFAPs and MUAPs, and stores their pre-calculated values.

- *Force model* class receives *Motor unit pool* object as an argument, it generates and stores twitch forms, simulates the force in the function of passed spike trains and encapsulates nonlinearity compensation model.

- *Profile* class permits to set up the parameters of contraction profile, generate its form, differentiate the recruitment, de-recruitment, no-contraction, and steady contraction phases.

Despite the fact that an effort has been made to separate the model implementation from its usage by encapsulating its parts into classes, it is still hard to use for a novice. If the community shows interest in this model, we will try to further optimize it and provide a graphical user interface for it.

5.4.3 Sub-sampling of the force

As we have already mentioned, the simulation model works at two different sampling frequencies: f_{sl} and f_{sh} (f_s "low" and "high"). Former is used for the contraction profile setting, nonlinearity compensation, PID control, and the output force, while the latter is for SFAP, MUAP and EMG modelling.

The necessity to down-sample the force control part comes from the fact that force as a signal lies in a narrow low-frequency spectral band that is bounded by approximately 10 Hz. However, down-sampling this process to 20 Hz would introduce a delay of at least 50 ms between the establishment of new force output value and generation of new compensatory command. Sampling this part at f_{sh}=20 kHz makes the coefficients of PID controller and of the linear model of excitation-force relationship hard to interpret. Thus, we have chosen a value of 50 Hz that gives an acceptable trade-off between the two effects.

The parts of the simulation model that run at different frequencies are illustrated in Figure 5.8. That is, green color indicates the part of the system that runs at the low sampling frequency, blue at the high frequency and grey is for the motor neuron pool model that generates spike trains with continuous values of time instants. The last holds if firing timings are generated using (5.2), while in case of (5.4) the timings are discrete simulated at f_{sh}.

5.4.4 Incremental computation

Since the output of the force model is used as the feedback to the PID controller, the excitation becomes a function of output force. Therefore, the firing instances generation, EMG and force models should be implemented in a way that permits

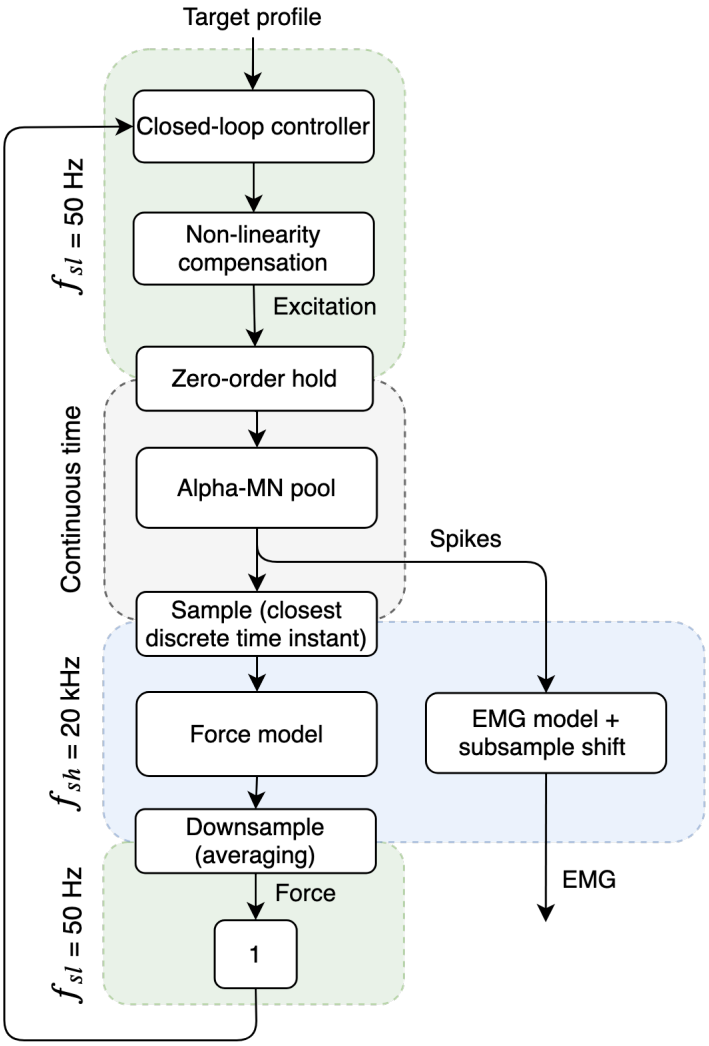


Figure 5.8: Schematic representation of the parts of the simulation model that run in continuous time as well as in high (f_{sh}=20 kHz) and low (f_{sl}=50 Hz) sampling frequencies. Zero-order-hold upsampler and averaging downsampler are used.

incremental computation of their outputs. That is, the state of these models should be retained between the calls on subsequent frames of their inputs.

Firing instants generation model proposed above naturally fits this requirement, saving only vector $T[n] = \{T_1[n], T_2[n], \ldots, T_n[n]\}$ of MN sojourn times between the calls. Force and EMG models involve the convolution of spike trains with corresponding waveforms, which implementation in an incremental way is also straightforward. To speed up the calculation that does not require the feedback control, these models have also normal fully vectorized implementations.

5.4.5 Data-set generation

The most computationally expensive part of the entire simulation model is the pre-calculation of SFAPs using equation (4.14) and the sub-sample shift of SFAPs during the generation of new realizations of MUAPs. The latter can be skipped in simulations that do not require precise EMG modeling (e.g., simulation of neuromuscular jitter and sub-sample firing instants). On the other hand, the former is necessary and should be done at least once in order to establish the EMG model. That is, the SFAPs can be pre-calculated and stored for future use.

Once, the model is set up, EMG and force signals are generated in the function of the target profile specified by the user. The system automatically names the output files for the signals, generates annotation, MUAP dictionary and stores them in a separate folder. To automate the data-set generation process, this functionality can be wrapped into a loop that goes through the contraction profiles. The model can be off-loaded from the Matlab memory and uploaded again from a file.

5.5 Conclusion

In this chapter, we have provided a step-by-step description of our muscle contraction simulation model. First, a model of excitation-rate curve was introduced, permitting to define the instantaneous firing rate of MN as a function of excitation. Next, we described the spike train generation approach based on hazard rate function and Markov chains modulated by the instantaneous firing rate. Then, classic twitch and total contraction force models were presented. That is, the total force is a sum of spike trains convoluted with twitches and weighted by a non-linear rate-dependent coefficient. Since our objective was the generation of force following a target profile, the PID controller and the compensation of the excitation-force relationship nonlinearity were introduced. In order to tune the PID contoller, blocks transforming the excitation into force were altogether modeled by a first-order low-pass filter. Finally, we have covered the class structure of our implementation of this simulation model, as well as its specific features.

The exact values of parameters that we used for simulation of contraction data will be provided in Chapter 6. Additionally, we will describe the target profiles that we used to generate datasets with various contraction properties. Finally, this simulation model is capable of providing both training and testing data for intent

derivation algorithms presented in Chapter 3.

Bibliography

[1] P. Contessa and C. J. D. Luca, "Neural control of muscle force: Indications from a simulation model," *Journal of Neurophysiology*, vol. 109, pp. 1548–1570, Mar. 2013.

[2] C. J. De Luca and Z. Erim, "Common drive of motor units in regulation of muscle force," *Trends in Neurosciences*, vol. 17, pp. 299–305, Jan. 1994.

[3] D. Kernell, "The Limits of Firing Frequency in Cat Lumbosacral Motoneurones Possessing Different Time Course of Afterhyperpolarization," *Acta Physiologica Scandinavica*, vol. 65, no. 1-2, pp. 87–100, 1965.

[4] C. J. De Luca and P. Contessa, "Hierarchical control of motor units in voluntary contractions," *Journal of Neurophysiology*, vol. 107, pp. 178–195, Jan. 2012.

[5] A. J. Fuglevand, D. A. Winter, and A. E. Patla, "Models of recruitment and rate coding organization in motor-unit pools," *Journal of neurophysiology*, vol. 70, no. 6, pp. 2470–2488, 1993.

[6] E. Petersen and P. Rostalski, "A Comprehensive Mathematical Model of Motor Unit Pool Organization, Surface Electromyography, and Force Generation," *Frontiers in Physiology*, vol. 10, p. 176, Mar. 2019.

[7] J. Monsifrot, E. Le Carpentier, Y. Aoustin, and D. Farina, "Sequential Decoding of Intramuscular EMG Signals via Estimation of a Markov Model," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, pp. 1030–1040, Sept. 2014.

[8] A. J. Fuglevand, D. A. Winter, A. E. Patla, and D. Stashuk, "Detection of motor unit action potentials with surface electrodes: Influence of electrode size and spacing," *Biological Cybernetics*, vol. 67, pp. 143–153, June 1992.

[9] M. Al Harrach, *Modeling of the sEMG / Force Relationship by Data Analysis of High Resolution Sensor Network*. Theses, Université de Technologie de Compiègne, Sept. 2016.

[10] J. L. Dideriksen, D. Farina, and R. M. Enoka, "Influence of fatigue on the simulated relation between the amplitude of the surface electromyogram and muscle force," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 368, pp. 2765–2781, June 2010.

[11] G. Venugopal, P. Deepak, D. M. Ghosh, and S. Ramakrishnan, "Generation of synthetic surface electromyography signals under fatigue conditions for varying force inputs using feedback control algorithm," *Proceedings of the Institution of*

Mechanical Engineers, Part H: Journal of Engineering in Medicine, vol. 231, pp. 1025–1033, Nov. 2017.

[12] M. Cogliati, A. Cudicio, F. Negro, P. Gaffurini, L. M. Bissolotti, and C. Orizio, "Influence of age on motor control accuracy during static ramp contractions," *Experimental Brain Research*, May 2019.

[13] K. G. Keenan and F. J. Valero-Cuevas, "Experimentally Valid Predictions of Muscle Force and EMG in Models of Motor-Unit Function Are Most Sensitive to Neural Properties," *Journal of Neurophysiology*, vol. 98, pp. 1581–1590, Sept. 2007.

[14] S. R. Bowling, M. T. Khasawneh, S. Kaewkuekool, and B. R. Cho, "A logistic approximation to the cumulative normal distribution," *Journal of Industrial Engineering and Management*, vol. 2, pp. 114–127, July 2009.

[15] C. K. Thomas, R. S. Johansson, G. Westling, and B. Bigland-Ritchie, "Twitch properties of human thenar motor units measured in response to intraneural motor-axon stimulation," *Journal of Neurophysiology*, vol. 64, pp. 1339–1346, Oct. 1990.

[16] F. E. Zajac, "Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control," *Critical Reviews in Biomedical Engineering*, vol. 17, no. 4, pp. 359–411, 1989.

Chapter 6

Simulation and experimental results

In this chapter, we will illustrate and analyze the effectiveness of decomposition-based effect estimation models presented in Chapter 3. First, we will use a simulated data set, generated using the models in Chapters 4 and 5. Then, an experimental data set, acquired from first dorsal interosseous muscle during index finger abduction, will be presented and used for the testing.

Between the estimation models presented in Chapter 3, we have chosen the one based on the joint transition law (Section 3.3.6), as the most detailed and providing the best estimates of effect. In the following, we will refer to it as "Joint law" or as "proposed model". As a reference model, we pick the force estimator based on cumulative spike train, presented in [1], to which we will refer as "CST-based" or "reference model". To quantitatively compare the performance of both, we use the coefficient of determination R^2, a measure commonly used in the domain of prosthetic control [2, 1, 3, 4].

Tests on simulated and experimental data will be further divided into two setups: *offline* and *online* estimation. Term "offline estimation" will be applied when the decomposition and effect data are pre-simulated or pre-acquired, and the estimation is made *a posteriori*, in an open-loop fashion. That is, the result of the estimation is not fed back to the subject and does not affect the contraction process. On the contrary, "online estimation" will refer to the setup where the estimate is provided to the subject in real-time, thus permitting him or her to adjust the neural input to the muscle to reach a specific value of the estimated effect.

Offline setup is commonly used in prosthetic control studies. It is relatively simple, allows to reproduce the results, and permits to use the same data set in different approaches in order to compare them. However, it lacks the possibility for the subject to adapt to the control strategy for better performance. Online setup, on the contrary, involves immediate feedback of the results to the subject, providing him with the possibility of adjustment. We have established both setups using simulation models presented in Chapters 4 and 5. Our experimental study presented in this chapter, however, currently involves only offline setup. The online experimental setup, its current state, and future perspectives will be discussed at

the end of the chapter.

6.1 Simulated data set generation

Muscle geometry and motor neuron pool parameters

To select the parameters of the simulation model and make this simulation as close as possible to our experimental setup, we will try to approximate a real muscle that we used our experimental studies, which is first dorsal interosseous (FDI).

Considering multiple assumptions that were made while establishing this simulation model, precise simulation of muscle dynamics and bioelectric properties is not possible. However, we can use a number of muscle characteristics provided by experimental studies [5].

Table 6.1 presents the parameters used in the simulation. We try to support the entries of this table by a corresponding reference or example. In the entries where references are missing, we assign values that are, to our best knowledge, the most physiologically consistent.

Parameters of excitation-rate relationship

Parameters of the excitation-rate model are linear functions of the MN's size s_i (see Section 4.2.1). The decrease of minimal and maximal rates, as well as of the slope, is in accord with experimental studies [8]; the actual values of the parameters are within physiological range [9].

Parameters of innervation and terminal arborization geometry

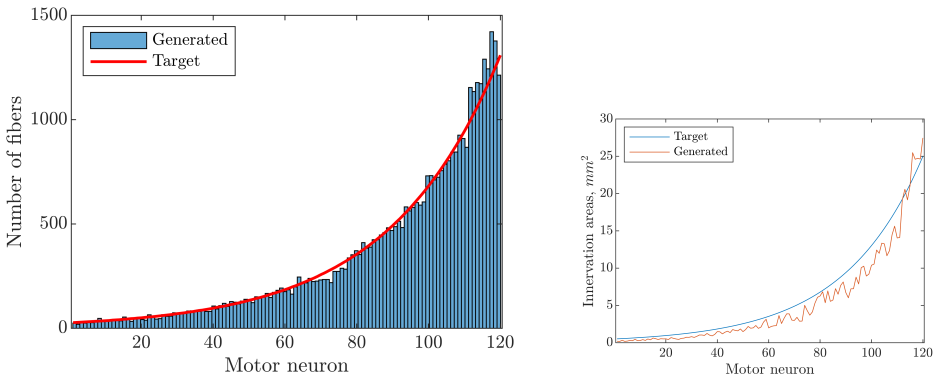
Parameters of innervation and arborization geometry are important only when actual forms of MUAPs is of interest. This is not the case in this simulation study, since decomposition of simulated iEMG does not take place, and simulated spike trains are directly used instead. However, the set MNs whose spike trains are included to the decomposition is defined by the amplitudes of MUAPs, according to criterion described in Section 4.5.2.

Parameters of innervation, such as the adjacency restriction parameter n_c and scattering coefficient C are adjusted in order to provide MU's innervation areas and numbers that are close to their model-imposed values (see Section 4.2.4 and Figure 6.1).

Parameters of terminal arborization geometry are tuned in such a way that they provide MUAPs with correct physiological lengths of 2.5 ms to 10 ms. Their values, as well as the parameters of innervation geometry, are presented in Table 6.1

Electrode simulation parameters and contraction profiles

The electrode was simulated as two observation points, arranged in a differential configuration. The points were placed in xy (cross-sectional) plane of the muscle, in its center, with 1 mm of the distance between the two.



(a) Innervation numbers of simulated MUs

(b) Innervation zone areas of simulated MUs

Figure 6.1: Simulated innervation numbers and innervation territories areas, compared to their values imposed by the model.

Contraction profiles, used for training and testing data simulation, mimic typical profiles used in experimental studies. That is, all contractions are limited to 25% MVC, the training data set contains two ramps with slopes of 5 and 10 % MVC/s, while the testing set consists of trapezoidal contractions reaching 25% MVC with slopes of 5 and 10 % MVC/s.

All simulations are performed at two sampling frequencies. The first, f_{sh}=10 kHz, is for iEMG simulation, while the second one, f_{sl}=50 Hz, applies to the force profile, force feedback and PID controller output calculation, as shown in Figure 5.8 in Chapter 5. These parameters are outlined in Table 6.1.

6.2 Experimental data set

Experimental iEMG was acquired from first dorsal interosseous muscle of a healthy 28 years-old man during the abduction of the index finger in the vertical plane. A needle electrode (30G Myoline disposable concentric needle) was inserted in the muscle belly at a depth of approximately 3 mm. iEMG was conditioned using the OTBioelettronica MEBA amplifier with band-pass filtering between 100 Hz and 4 kHz and sampled at a frequency of 10 kHz. The angle of abduction was acquired by an inertial measurement unit (IMU) MPU9250 attached to the tip of the index finger. The measured angle was shown to the subject on a PC screen. The rest of the hand, except for the index finger, was immobilized by a combination of foam rubber constraints.

Full abduction angle (around 35 degrees) was considered as the maximum contraction level, to which all contraction profiles were defined relative to. The subject was asked to abduct the index finger to match the measured angle with a specified profile. Three forms of profiles were used: slowly growing ramp, trapezoidal and constant level. Trials where the subject failed the matching, were rejected. Each profile lasted at most 25 seconds. One minute of the rest was given to the subject

between the acquisitions. IMU's harness was designed to be lightweight, the subject did not report any fatigue between the trials or at the end of the acquisition.

In total, two ramp profiles reaching 50% of maximum abduction angle (MAA), two trapezoidal profiles reaching 25% MAA and two constant profiles at each level of contraction between 10 and 50% MAA with a step of 10% were acquired. The iEMG data was decomposed by hand using EMGLAB software, annotations were aligned in such a way that each MU had the same index in different trials.

6.3 Evaluation methods

6.3.1 Criterion of estimation quality

For comparison with existing methods, we will use the joint model based on recruitment and IPI distribution described in Section 3.3.6. Corresponding estimates will be referred to as "Joint law" or "proposed model". Estimates provided by an existing reference model will be denoted as "CST-based". The reference model will be described in the following section.

In order to compare the effect estimates by the proposed and reference models, we will provide corresponding plots, using testing simulation and experimental profiles. As a numerical measure of fit, we will use the coefficient of determination R^2. It is calculated as a portion of the effect variation that is explained by the estimate:

$$R^2 = 1 - \frac{\sum_n (e[n] - \hat{e}[n])^2}{\sum_n (e[n] - \frac{1}{N}\sum_n e[n])^2} \tag{6.1}$$

where $e[n]$ is n-th effect sample from the contraction data, $\hat{e}[n]$ is its estimate provided by the estimation model, based on the decomposition data, N is the total number of samples.

The maximal value of R^2 is 1, which appears in the case of perfect estimation, i.e. when the mean square of the estimation error is equal to zero.

6.3.2 Reference effect estimation model

In this and in the following sections, the proposed effect estimation model is compared to an existing decomposition-based estimation approach, proposed in [1]. It consists of summing up the decomposed spike trains in one vector, called cumulative spike train (CST). Then, it is convoluted by an impulse response of a second order critically damped linear system, representing a generalized MU twitch [7]. In fact, this approach approximates the linear model of total muscle contraction force (5.6), using one single twitch form and decomposed spike trains:

$$\hat{e}(t) = \sum_{i=1}^{N} \sum_{k=1}^{|I_i|} \tau(t - I_{i,k}) \qquad \tau(t) = \frac{P_\tau}{T_\tau} \cdot t \cdot \exp(-t/T_\tau) \tag{6.2}$$

where $I_{i,k}$ is the time of k-th spike in i-th MN's spike train and $\tau(t)$ is the twitch response.

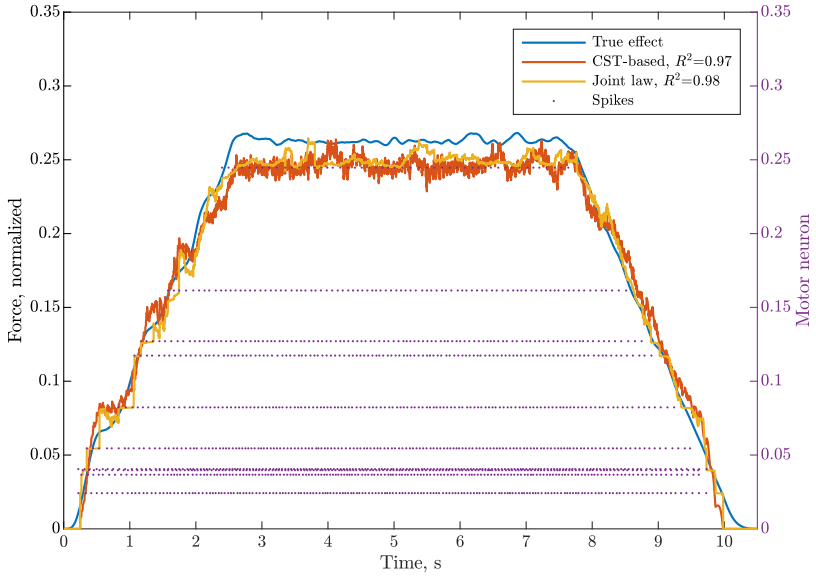


Figure 6.2: Estimates provided by CST-based and joint estimation models, on the simulated trapezoidal force profile reaching 25% MVC. Spikes of MNs are traced on their estimated recruitment threshold mean values λ (see Section 3.2.3).

Parameters P_τ and T_τ are identified by minimizing the sum of squares of the errors between the observed effect and its estimate. This procedure will be performed in each of the following examples, using the same training data as for the presented estimation model.

6.4 Offline effect estimation for simulated data

Using the simulation model with parameters provided in 6.1, we have generated a trapezoidal profile reaching 25% MVC. Both CST-based and the proposed model were identified on the same training set (two ramp profiles with 5 and 10 % MVC slopes reaching 25% MVC). In total, ten MNs were considered decomposable for this contraction and included in the decomposition.

Figure 6.2 shows the outputs of estimation models. As we can see, both are visually very close to the true value of the effect, providing high R^2 measures. This result shows that both methods can perform equally good when the number of decomposed spike trains reaches an approximate limit of current real-time decomposition algorithm capability [10] (see Section 1.3 for details).

Let us assess their performance on a reduced number of decomposed MNs. Comparing to the presented simulation, the number of decomposed MNs can be smaller due to an increase of additive noise or the use of an electrode with a smaller acquisition area. In order to assess the estimators' performance in such cases, we will generate random subsets of these ten MNs and evaluate R^2 for the estimates produced using the corresponding spike trains. More specifically, we have generated all the possible 1023 combinations of ten MNs and ran the estimation on a ran-

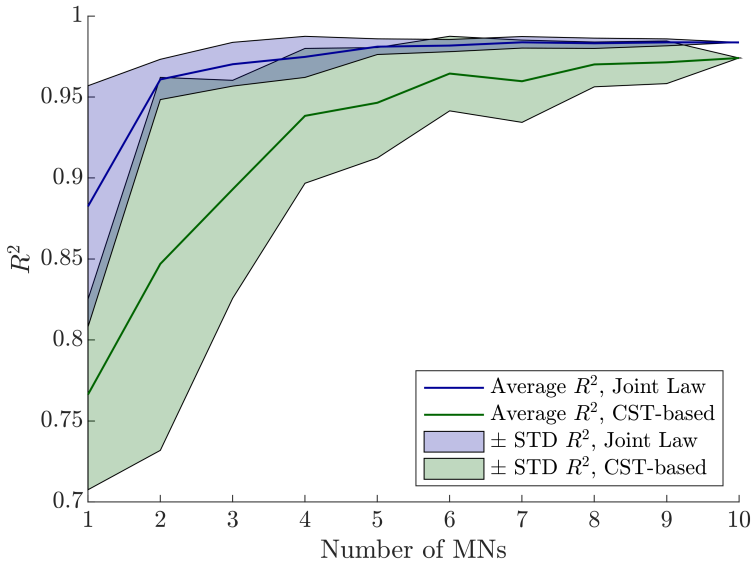


Figure 6.3: Average values and standard deviations of R^2 measure for the estimations performed by the proposed and CST-based models. Estimations are made using spike trains of random subsets of the original set of decomposable MNs. R^2 values are averaged across subsets with the same sizes, i.e., numbers of included MNs.

dom subset of 256 of them. We then ran the estimation for all the possible missing combinations of one and two MNs.

Figure 6.3 demonstrates the mean and standard deviation of the R^2 calculated for different numbers of MUs included in the subset, for both estimation models. One can see that the proposed model provides significantly better results for smaller numbers of decomposed MNs, both in terms of the average R^2 value and its standard deviation. With the growing number of MNs, both models converge to approximately R^2=0.95.

Both models profit from larger numbers of decomposed spike trains. As we have noted earlier, CST-based estimation is an approximation of total force formula and therefore works better when it is provided with more spike trains. The proposed model efficiently merges the information from multiple spike trains as well, being also able of properly weighting their contributions when provided with small numbers of MNs. CST-based model, on the contrary, considers MNs to exhibit similar contributions to the effect production, thus suffering from the cases when small numbers of decomposed MNs have very different effect-rate curves.

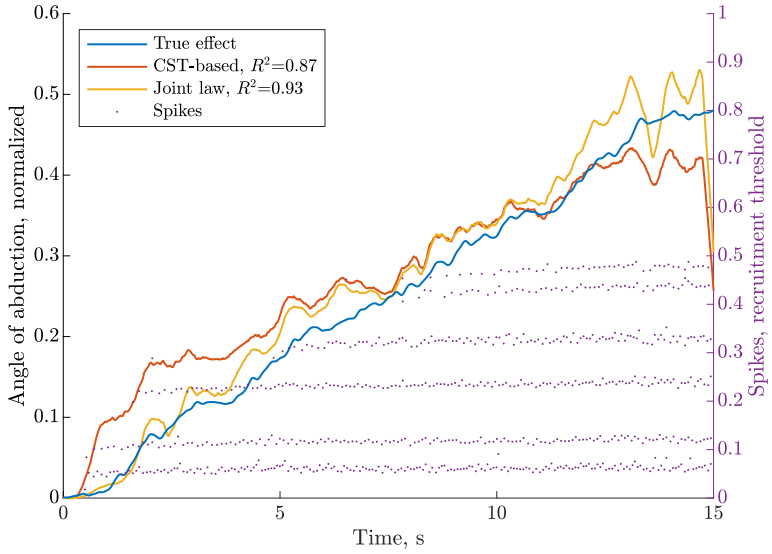


Figure 6.4: Estimation of abduction angle during experimental ramp contraction reaching 50% of maximum angle. Estimates are provided by the proposed model (Joint law) and by CST-based reference model.

6.5 Offline effect estimation for experimental data

6.5.1 Ramp contractions

At first, we will train and test the estimation models on two different trials of the same ramp contraction profile reaching 50% MAA. Figure 6.4 presents the results of testing data. Estimates provided by the two models are highly covariated (coefficient of covariation 0.97) due to the occasional drops in the instantaneous firing rates of MNs. The proposed model shows a better fit in the lower range of contractions. Similar analysis on random subsets of MNs as in Section 6.4 is provided in Figure 6.5. One can see that, in this setup, the proposed model performs consistently better than CST-based.

6.5.2 Trapezoidal and constant contractions

Now let us use a different setup, in which two ramp contractions constitute training data set, while a trapezoidal contraction and three constant contractions at different levels are used for testing. This setup is more challenging because testing and training profiles have now different slopes (3.5 degrees per second for the ramp, 2.5 degrees per second for the trapezoidal, vertical slope for constant contractions). Moreover, profiles now contain constant and decreasing effect intervals.

Results for the trapezoidal testing profile are shown in Figure 6.6. As we can see, both models fail to correctly fit the decreasing effect. Overlaid spike trains show that MNs' behavior on the negative slope is different from that on the positive slope, e.g.: MNs tend to de-recruit at a higher level of the effect. Additionally, 2nd MN, whose mean RT, estimated from ramp contractions, is lower than that of the 3rd

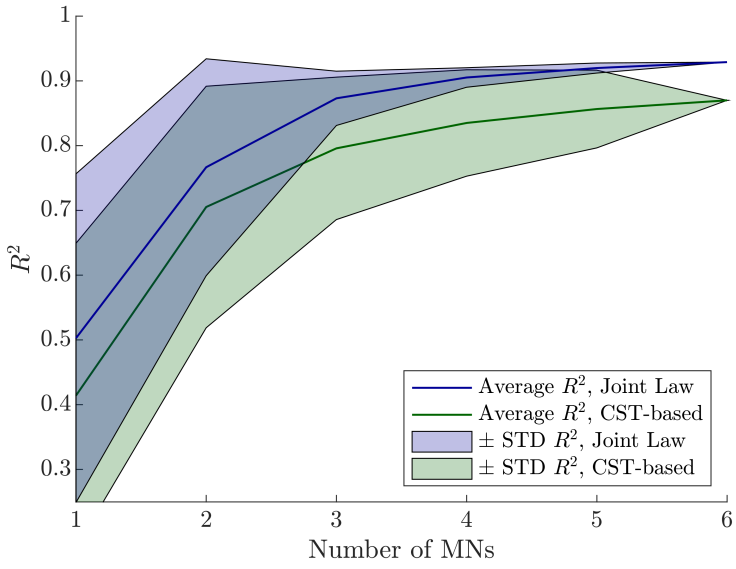


Figure 6.5: Average values and standard deviations of R^2 measure for the estimations performed by the proposed and CST-based models on experimental ramp data set. Estimations are made using spike trains of random subsets of the original set of decomposable MNs. R^2 values are averaged across subsets with the same sizes, i.e., numbers of included MNs.

MN, de-recruits faster, which is inconsistent with the proposed model.

Results for constant testing profile at 10%, 20% and 30% of maximum abduction angle (MAA) are shown in Figure 6.7. Both models successfully detect the onset of the movement, providing, however, very noisy estimates of the effect at its plateau. This is due to the small number of spike trains used for estimation (correspondingly 3 and 4 for contractions at 10% versus contractions at 20% and 30% MAA).

6.6 Online simulated effect estimation

6.6.1 Virtual subject

In this section, we will present and discuss the results of the estimation in a closed-loop setup. That is, the effect estimation will be integrated into the force simulation model described in Chapter 5. This setup simulates the prosthetic control by an amputee, when the biomechanical end-effector, as well as the effect it produces, are missing. Instead, the amputee is provided with an estimate of the effect made by a decomposition-based estimation model. In this section, we aim to study the adaptation and closed-loop compensation capability of a simulated amputee user in cases of the proposed and the reference estimation models.

Figure illustrates the used approach. A model of an intact subject is presented in 6.8(a), we use it to generate the training data for effect estimation model. In a real experimental setup, this procedure could be realized using either mirrored bilateral

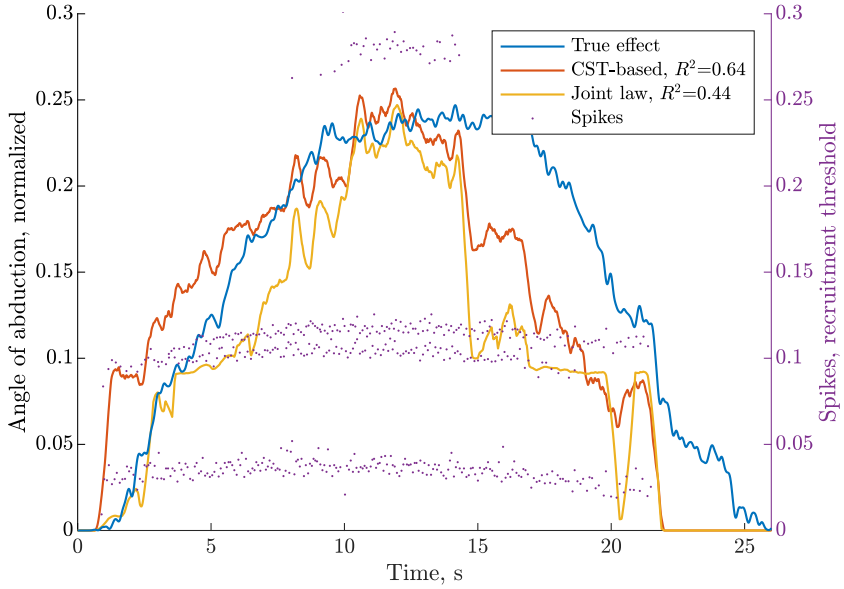


Figure 6.6: Results of angle estimation for experimental trapezoidal contraction. Spiking behavior of MNs differs for negative and positive slopes, causing both models to provide poor estimates of the angle.

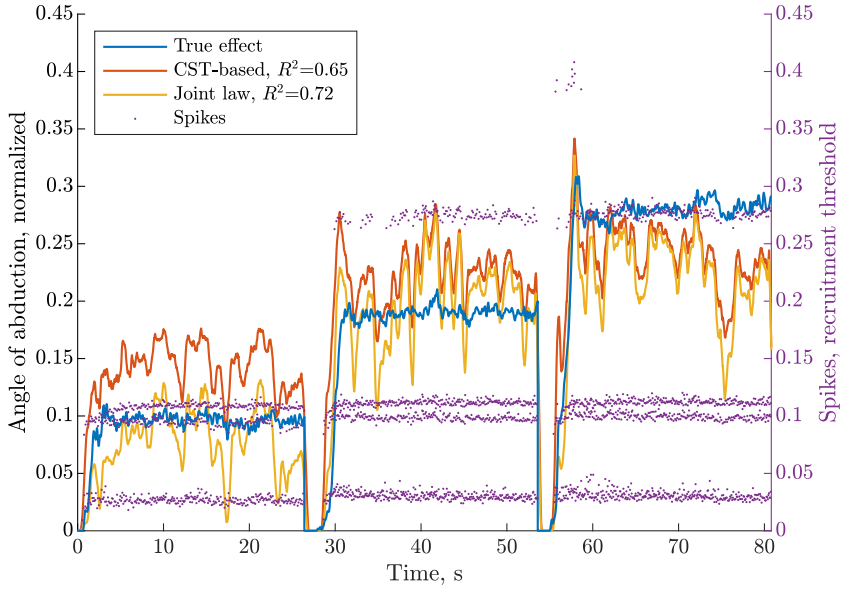


Figure 6.7: Results of angle estimation for experimental constant contraction profile. Both models correctly detect the onset of the movement, providing, however, noisy estimates at its plateau due the small number of available spike trains.

training or prosthesis-guided approach. Next, model of an amputee is presented in 6.8(b). That is, now the muscle block and the end-effector are missing, while a decomposition-based estimation model is added. The estimate of the effect produced by this model becomes the target variable that the virtual subject seeks to control by following a specified profile.

We will use the same simulation model parameters as in the previous, offline, case. Also, the same pre-simulated ramp contraction data will be used to identify the estimation models. Next, we will estimate the static nonlinearity of the estimate by measuring it during a slow linearly increasing ramp contraction. This nonlinearity will be approximated by a 6-th order polynomial curve. Next, similarly to Section 5.3.3, we will tune a PID controller that will adjust the excitatory input to the MN pool in order for the estimator to output a specified effect value. The updated blocks are indicated by blue color in Figure 6.8.

6.6.2 Results

First, we simulate a close-loop control using all the ten available MNs. Corresponding results are present in Figure 6.9. Both models provide very consistent estimation, as in the offline setup (see Figure 6.2).

Next, let us apply the same analysis method as in previous cases: we will generate all possible combinations of these ten MNs in the decomposition, rerun the simulation for each case and calculate the corresponding R^2 measures. The PID controller coefficients, calculated for the case of all ten MNs, were left constant across the simulations. However, the nonlinearity pre-compensation was recalculated for both models in each new MNs subset to adapt to the new spiking behavior (especially necessary for the CST-based model, as we have seen in the previous section). The CST-based model was re-identified for each new MN subset.

Results of this test are presented in Figure 6.10. As we can see, the proposed model provides a better average R^2 with less variation for all numbers of MNs in a subset.

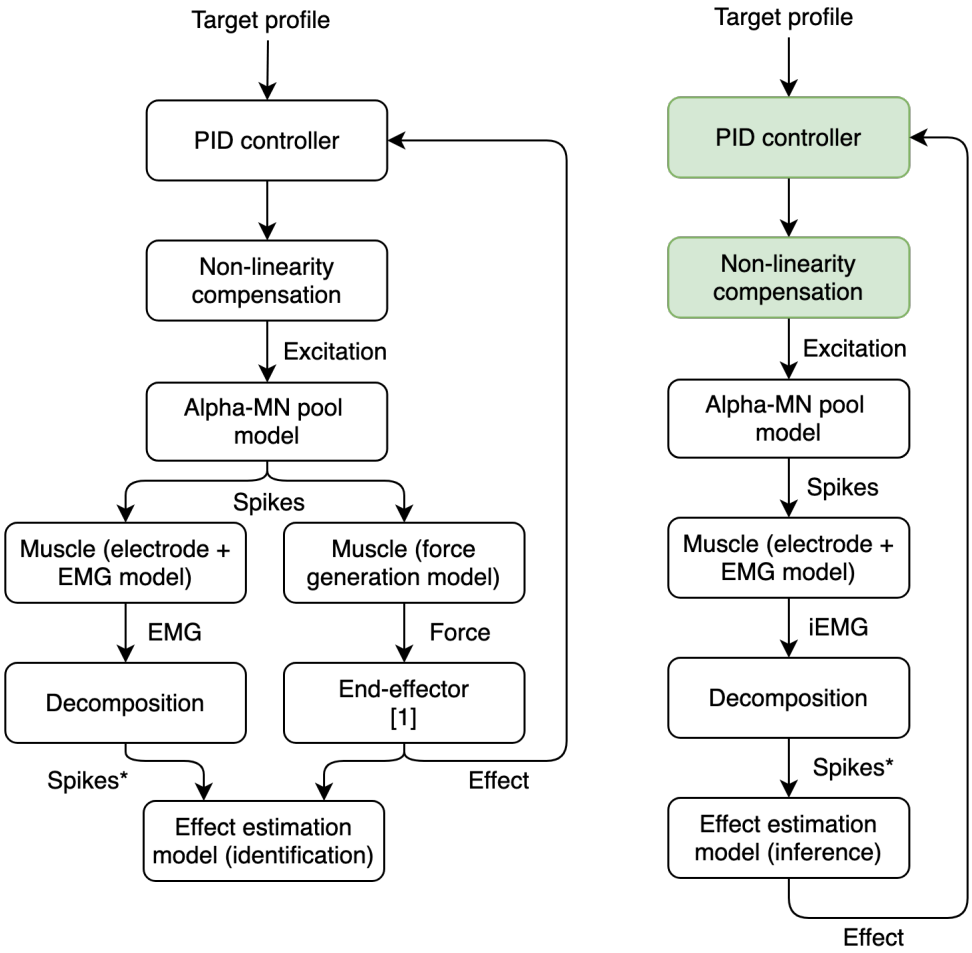


Figure 6.8: Schematic representation of the simulation model in cases of training (a) and testing (b) of the effect estimation model for online setup. (a) - Effect and decomposition data are generated simultaneously during a contraction with specific profile. Corresponding decomposition and effect data are used to identify the effect estimation model. (b) - The effect simulation part of the model is now missing, decomposition is passed to the estimation model to infer the effect. The output of the estimation model is directed to the controller that seeks to compensate for the possible errors. *Spikes** signifies decomposed spike trains, small subset of the entire MU pool. Green block represents the parts of the model that should be re-calibrated after replacement of the effect simulation by effect estimation.

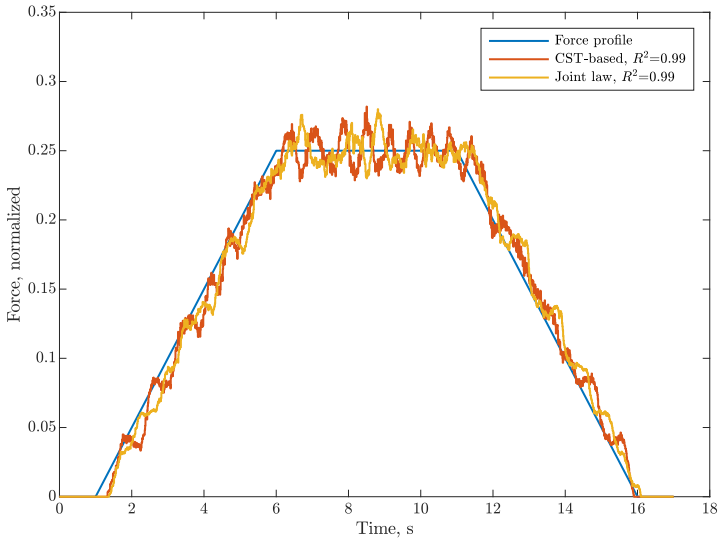


Figure 6.9: Results of estimation for a simulated trapezoidal contraction in online setup.

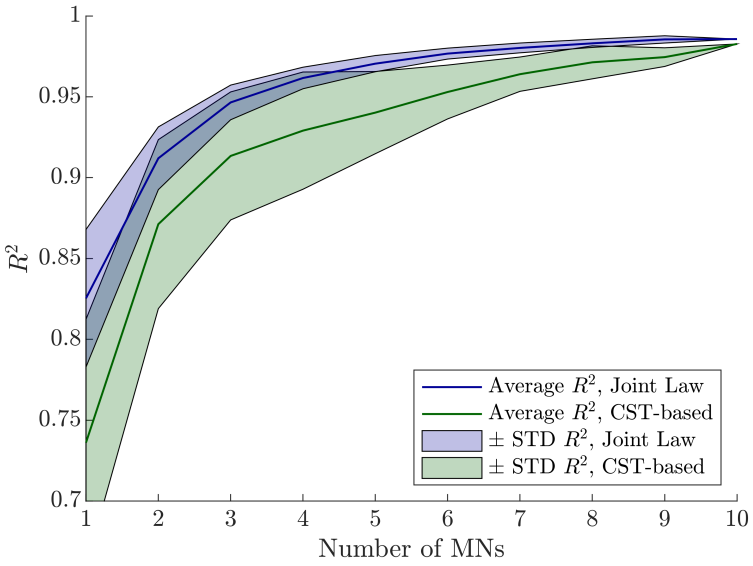


Figure 6.10: Average value and variance of R^2 measures as functions of resulting number of MNs included in the decomposition, online simulated setup.

6.7 Online experimental effect estimation: current state and perspectives

In our recent study [10], we have implemented recursive decomposition algorithm [11] and have achieved real-time decomposition of iEMG. More specifically, experimental signals containing up to eight MUs were decomposed with a precision exceeding 95%, and with a maximum delay of 70 ms, using GPU Nvidia Tesla K80. However, in the corresponding tests, the online setup was simulated by frame-by-frame processing of pre-acquired signals. A real experimental setup, on the contrary, would require the acquisition and processing of a signal to be performed simultaneously.

In this section, we will describe our approach to this problem, as well as our pilot setup for an experimental study of real-time decomposition implementation. Unfortunately, we could not yet perform an online decomposition-based effect estimation. However, as we will show later in this section, the established decomposition system is functional and can be used for this purpose in the near future.

6.7.1 Description of the real-time decomposition system

As an iEMG acquisition system, we used MEBA by OTBioelettronica. The corresponding Windows application, OTBiolab, continuously publishes the iEMG and auxiliary data on a specified local port of the PC. This software package also contains a Java class that can be used in Matlab to access these data frames. Thus, we created a Matlab script that transmits the iEMG data to the GPU and receives back the decomposition results for further visualization and processing.

The problem of data transmission was complicated by the fact that the selected GPU was physically located on a remote computer, accessible via the local network. Thus, we have decided to transmit the iEMG and decomposition frames using TCP/IP protocol. Since the remote computer accepted only encrypted connections through Port 22, we have used a so-called SSH-tunnel, which virtually connects two local ports on two different devices via Port 22. Matlab (on the acquisition computer) and the decomposition software (on the remote computer) interacted their local ports using TCP/IP sockets (*tcp* function in Matlab and *iostream* library in C++). The structure of this approach is illustrated in Figure 6.11.

The size of the data frame was chosen to be 50 ms, to ensure the smoothness of decomposition-based estimates. Due to the nature of the used real-time decomposition algorithm, spike trains in older frames could be corrected using iEMG data in the newer ones. Thus, a longer window (250 ms) of the most recent decomposition results was transmitted within each frame.

In the test runs, the established system has shown real-time decomposition results similar to the ones obtained in [10], while transmitting the same experimental signal in a frame-by-frame mode from the acquisition PC to the remote GPU. The maximal lag between newly acquired iEMG signal and its decomposition was approximately 150 ms after the recruitment of eight MUs.

A Matlab-based GUI was also developed to control the acquisition parameters,

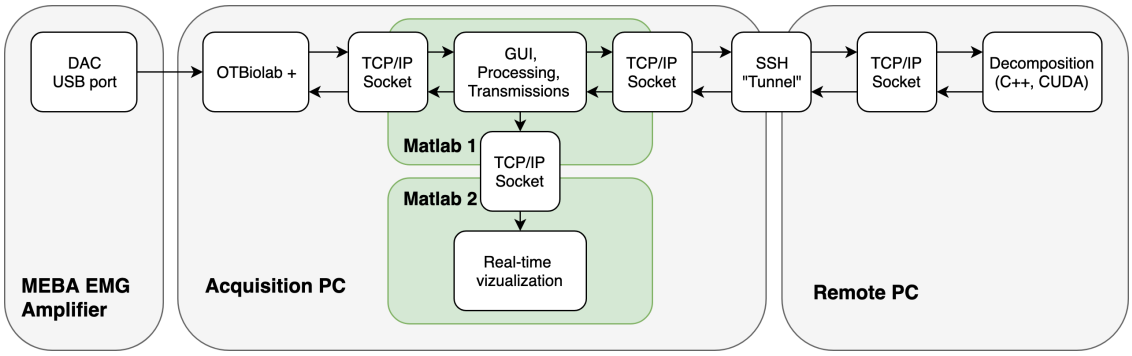


Figure 6.11: Structure of the real-time decomposition system. Gray squares represent physically separate devices, green squares represent independent Matlab sessions.

observe the signal and its decomposition in real-time, and to provide the visual feedback to the subject. A technical detail worth mentioning: iEMG signal acquisition, communication with GPU and further processing were running in a single Matlab session, while the visualization was running in a separate one. Although the same computational power is used, this two-sessions setup provides much faster plotting that a normal one. A short video illustrating the functioning of the system is provided[1].

6.7.2 Pilot setup for experimental real-time decomposition

As a next step, we have tested a setup in which real-time decomposition was provided for an iEMG signal acquired from the FDI muscle of an intact subject. The protocol was similar to the one described at the beginning of this chapter (see Section 6.2). Since it was a pilot study, the contraction was limited in order to have at most four active MUs at the same time.

First, a short portion of iEMG was decomposed by hand using EMGLAB to obtain initial estimates of MUAP forms. Then, MUAPs were shortened to 5 ms and re-scoped to make each MUAP start with a value significantly different from the level of the instrumentation noise. This procedure, as described in [11], increases the quality of decomposition. Then, the resulting forms of MUAPs were transmitted to the GPU.

The testing contraction was limited to a lower value to ensure the absence of non-identified higher threshold MUs in the iEMG. Correct real-time decomposition of three concurrently active MUs was finally obtained. Its quality was assessed using the residual signal after reconstruction. This result is purely qualitative and is reported to demonstrate the principal viability of the established real-time decomposition system.

Unfortunately, we could not yet conduct the tests that would involve online intent estimation in a real subject. However, results obtained so far show that it

[1] Real-time decomposition example.

is technically feasible in the near future. These results also suggest that the GPU needs not to be physically located in the same device that acquires the iEMG or uses the decomposition results. This is a promising feature for an application in prosthetics, since the use of GPUs in a wearable device is undesirable, due to their size and power supply requirements. TCP/IP protocol scales to other means of communication, such as the internet which widens the possible solutions.

6.8 Discussion and conclusion

In this chapter, we have described our simulation and experimental protocols, as well as the reference estimation model and a comparison criterion. The reference model was proposed in a recent study [1] and consists of smoothing a cumulative spike train by a twitch response [7]. The coefficient of determination R^2, commonly used in related studies, was used as a measure of the estimation quality.

Simulation data consisted of force observations combined with simulated spike trains of MUs considered detectable and decomposable in the given protocol. We have identified the parameters of both models using ramp contraction data. The force inference was tested on a trapezoidal contraction. Provided with the maximal number of decomposed MNs (ten), both models show good performance with $R^2 > 0.95$. Further, we analyzed the robustness of both models in the case of smaller numbers of available MNs, which have showed that the proposed model performs consistently better on such reduced inputs that the reference.

Experimental data consisted of manually decomposed iEMG of FDI muscle acquired during index finger abduction. Contrary to the simulation data, estimation models were predicting abduction angle instead of force. In the setup where both training and testing contraction data were of the same profile (ramp), both models provided correct estimates of the angle ($R^2 > 0.85$), with the proposed one being more robust to the reduction of the number of available MNs in the decomposition. Moreover, Figure 6.4 shows that the proposed model provides better fit at all levels of contraction, while the CDR-based model, treating all MNs as having similar contributions to the effect production, fits the data only at a specific level of contraction and fails at the others.

In the experimental setup where the models were identified using ramp contraction data and tested on a trapezoidal profile, the inconsistencies in the firing behavior of MNs during positive and negative slopes of the profile have significantly affected the performance of the both.

Bibliography

[1] C. K. Thompson, F. Negro, M. D. Johnson, M. R. Holmes, L. M. McPherson, R. K. Powers, D. Farina, and C. J. Heckman, "Robust and accurate decoding of motoneuron behaviour and prediction of the resulting force output: Neural drive to muscle and resulting force output," *The Journal of Physiology*, vol. 596, pp. 2643–2659, July 2018.

[2] H. Yeganegi, Y. Fathi, and A. Erfanian, "Decoding hind limb kinematics from neuronal activity of the dorsal horn neurons using multiple level learning algorithm," *Scientific Reports*, vol. 8, Dec. 2018.

[3] M. Kim and W. K. Chung, "Spatial sEMG Pattern-Based Finger Motion Estimation in a Small Area Using a Microneedle-Based High-Density Interface," *IEEE Robotics and Automation Letters*, vol. 3, pp. 234–241, Jan. 2018.

[4] D. Farina, A. Holobar, R. Merletti, and R. M. Enoka, "Decoding the neural drive to muscles from the surface electromyogram," *Clinical Neurophysiology*, vol. 121, pp. 1616–1623, Oct. 2010.

[5] B. Feinstein, B. Lindegård, E. Nyman, and G. Wohlfart, "Morphologic studies of motor units in human muscles," *Cells Tissues Organs*, vol. 23, no. 2, pp. 127–142, 1955.

[6] A. Hamilton-Wright and D. Stashuk, "Physiologically Based Simulation of Clinical EMG Signals," *IEEE Transactions on Biomedical Engineering*, vol. 52, pp. 171–183, Feb. 2005.

[7] A. J. Fuglevand, D. A. Winter, and A. E. Patla, "Models of recruitment and rate coding organization in motor-unit pools," *Journal of neurophysiology*, vol. 70, no. 6, pp. 2470–2488, 1993.

[8] C. J. De Luca and P. Contessa, "Hierarchical control of motor units in voluntary contractions," *Journal of Neurophysiology*, vol. 107, pp. 178–195, Jan. 2012.

[9] K. G. Keenan and F. J. Valero-Cuevas, "Experimentally Valid Predictions of Muscle Force and EMG in Models of Motor-Unit Function Are Most Sensitive to Neural Properties," *Journal of Neurophysiology*, vol. 98, pp. 1581–1590, Sept. 2007.

[10] T. Yu, K. Akhmadeev, E. L. Carpentier, Y. Aoustin, and D. Farina, "On-line recursive decomposition of intramuscular EMG signals using GPU-implemented Bayesian filtering," *IEEE Transactions on Biomedical Engineering*, 2019. Manuscript in revision.

[11] T. Yu, K. Akhmadeev, E. L. Carpentier, Y. Aoustin, R. Gross, Y. Pereon, and D. Farina, "Recursive decomposition of electromyographic signals with a varying number of active sources: Bayesian modelling and filtering," *IEEE Transactions on Biomedical Engineering*, pp. 1–1, 2019.

Table 6.1: Simulation parameters.

Muscle geometry and motor neuron pool	
Number of muscle fibers (MFs), F	40000 [5]
Density of MFs, D_m	400 per mm^2 [6]
Muscle radius	$\sqrt{F/\pi D_m}$=5.65 mm
Muscle (and fibers) length, L	30 mm
Number of MNs, N	120 [5]
Magnitude of RT distribution, R	50
Maximal RT, M	0.75 (in normalized excitation)
Parameters of the excitation-rate curves [7]	
Minimum firing rate, $\varphi^{\min}$	$\varphi_i^{\min} = -5 \cdot s_i + 10$
Maximum firing rate, $\varphi^{\max}$	$\varphi_i^{\max} = -10 \cdot s_i + 40$
Slope of excitation-rate line, φ^{sl}	$\varphi_i^{\text{sl}} = -20 \cdot s_i + 50$
Coefficient of variation of IPI, C_v	1/8
Innervation and terminal arborization	
Adjacency restriction parameter, n_c	5 (see Section 4.2.4)
Scattering coefficient, C_s	inv-$\chi^2(0.99, 2) = 9.21$ (see Section 4.2.4)
NMJ clusters centers scattering parameters	$a_\mu = 1.5$, $b_\mu = 4$ (see Eq. (4.8))
NMJ within-cluster scattering parameters	$a_\sigma = 1.5$, $b_\sigma = 4$
MNAP velocity of an axon branch	$v_b = 5$ m/s
MNAP velocity in terminal arborization	$v_t = 2$ m/s
Neuromuscular jitter standard deviation	0.025 ms [6]
Sampling frequencies	
EMG and twitches sampling frequency, f_{sh}	10 kHz
Contraction force sampling frequency, f_{sl}	50 Hz
Training and testing data sampling frequency	1 kHz
Spatial sampling step along z-axis,	0.5 mm

Conclusion and perspectives

Results

The aim of this study was the establishment and evaluation of decomposition-based prosthetic control. This was to be done with respect to the limit of number of simultaneously decomposed spike trains provided by existing real-time decomposition approaches. Simulation and experimental studies were conducted in order to evaluate the proposed intent estimation model and compare it to an existing one.

Development of the simulation model pursued two main goals: generation of decomposition results similar to the ones observed in experiments and simulation of neural command to the contracting muscle. The first goal involved the modeling of motor neuron innervation zones, of the electrode geometry and motor unit action potentials. As an additional result, we have obtained a simulation model for multichannel iEMG with a possibility to dynamically vary the location of the electrode and model its effect on the acquired signal. We have provided several examples of how this model can be used to emulate existing studies on motor neuron innervation territories and multichannel decomposition. Finally, modelling of motor unit action potentials permitted to simulate the experimental decomposition, i.e. to automatically decide which MUs would be present in the final decomposition of the iEMG signal and thus generate simulated data sets.

Simulation of neural command to a contracting muscle involved the use of classic models of motor neuron firing behavior and of muscle contraction force, as well as closed-loop control and nonlinearity compensation. This model permitted to generate spike trains of motor neurons in the function of a specified force profile. That, in its turn, was used to generate data sets for evaluation of intent estimation models.

While developing the proposed intent estimator, we pursued a model-based approach. That is, we have used existing relations between the instantaneous firing rate and currently produced mechanical effect in order to infer the latter from the former. Three different approaches to this inference problem were proposed: one is based purely on the fact that certain motor neurons are recruited or de-recruited, the second is based on the inter-pulse interval statistics as a function of the effect, while the third approach merges the first two. As the tests on simulation and experimental data show, this last approach provides consistently better estimations than the reference model. We have also shown that, compared to the last, the presented approach shows better performance when provided with reduced number of

concurrently decomposed spike trains.

Perspectives

Intended effect estimation

The model-based approach to the estimation of intended effect proposed in Chapter 3 can be further modified within a number of principal directions.

RT-based cost function. As we have mentioned in Section 3.2.6, the recruitment-based effect estimation model provides a cost function that penalizes the effect estimates that are inconsistent with the current activation vector and the recruitment threshold distributions of the MNs. When not penalizing, this cost function is flat, so that the position of its minimum can be easily adjusted by an additional term in its expression. Such term may be a continuous function of the effect (e.g., sEMG envelope), which will permit to produce smooth estimates of the effect instead of the discrete ones presented in Section 3.2.6. These estimates will be still regularized due to the penalization effect of the effect-recruitment model.

Iterative inference for the joint model. Also, in this work we could not provide an iterative effect estimation for the joint model presented in Section 3.3.6. An iterative approach, analogous to the one presented in Section 3.3.4, is of interest since it will significantly accelerate the effect inference when using the joint estimation model.

Integration with the decomposition algorithm. The joint effect estimation model (see Section 3.3.6) and the decomposition algorithm we used in this work [1] are both based on hidden Markov modeling (HMM) and both operate with similar variables, such as motor neuron sojourn time $T[n]$, activation $A[n]$ and IPI distribution parameters $\Theta[n]$. However, while in [1] the hidden variables are $\Theta[n]$, the proposed model has a single hidden variable, which is the intended effect E.

These two HMMs can be merged, which may be of interest due to the following potential outcomes. First, instead of estimating $\Omega[n]$ of each MN, the decomposition algorithm will need to estimate only one hidden variable, which is the intended effect E, thus reducing the computation requirements. Second, Bayesian filtering of the $E[n]$, taking place in this case, may provide better estimates of the effect than the model proposed in this work. Third, probabilities of MNs activation will become functions of the intended effect, instead of being defined as constants, as it is currently done in [1]. This also has a potential to reduce the calculation requirements of the decomposition algorithm due to the pruning of the decomposition scenarios that are inconsistent with the recruitment model.

Possible approach to the estimation at higher contraction levels. All of the three proposed estimation models produce the estimates of the effect by accumulating information from spike trains of all available MNs. Inclusion of a new MN to decomposition (or exclusion of an existing one from it) will affect the precision of the estimate, but not the very possibility to produce it. Thus, these models will be viable in a hypothetical setup where a decomposition algorithm, in

order to not exceed the maximum possible number of concurrently decomposed MNs, excludes the least important ones from the decomposition. Naturally, importance of a MN can be defined by its contribution to the energy of EMG signal, relatively to the contributions of other active MNs. The MN that contributes the most at low contraction level can be completely masked by the larger ones at higher forces. Since most of the decomposition algorithms seek to minimize the residuals between the observed EMG and its reconstruction from decomposition, the contribution of such MN may become too insignificant to keep the track of it. The reduction of the computational complexity of the decomposition procedure after the exclusion of such MN from the model will outweigh the resulting minor loss of precision. Such an approach can be a solution to the limitation of low contraction forces in the existing real-time decomposition algorithms, while the proposed estimation models are capable of producing the effect estimates using corresponding decomposition data.

iEMG simulation model

Additionally to the applications of the intramuscular MUAP and EMG simulation model presented in this work, there potentially can be more of them. Let us provide a number of examples.

Simulation of myopathic signals. Myopathic iEMG signals are characterized by abnormally high firing rates of MNs, which is the result of reduction of the total number of functioning muscle fibers and, as consequence, of the total force produced by the muscle. This effect can be simulated by decreasing the amplitudes of the MU twitch responses proportionally to the their losses of muscle fibers. The feedback controller, facing the lack of produced force, will increase the excitation which will entail the increase of firing rates.

Simulation of neuropathic signals. Neuropathic iEMG signals are characterized by the presence of large complex MUAPs, which result from the re-innervation of denervated muscle fibers by the adjacent MUs. This process can be simulated by the choice of a random subset of MUs with subsequent redistribution of their muscle fibers among other MUs, according to the probabilistic procedure described in Section 4.2.4. Branch assignment can be performed based on the distance to the closest centroid originally identified by *k-means* algorithm. Locations of neuromuscular junctions can be re- drawn from the distributions in corresponding branches of the new MU.

Evaluation of approaches to MU territories estimation. As it is shown in 4, the estimation of MN innervation territory based on the amplitudes of its MUAP in different channels will produce poor results when the electrode does not pass through the center of the territory. As the simulation has shown, the farther the electrode is from the territory center, the lower its amplitude is and the lesser amount of high-frequency components it contains. Comparative analysis of MUAPs acquired from different MNs may give an approximate span of these frequencies. Estimation of the territory diameter can be thus based not only on the amplitude of the corresponding MUAP in different locations of the muscle, but also on its

frequency content. The simulation model proposed in this work can be used to test and validate such estimation approaches.

Bibliography

[1] T. Yu, K. Akhmadeev, E. L. Carpentier, Y. Aoustin, R. Gross, Y. Pereon, and D. Farina, "Recursive decomposition of electromyographic signals with a varying number of active sources: Bayesian modelling and filtering," *IEEE Transactions on Biomedical Engineering*, pp. 1–1, 2019.

UNIVERSITE BRETAGNE LOIRE MATHSTIC

UNIVERSITÉ DE NANTES

Titre : Modèles probabilistes fondés sur la décomposition d'EMG pour la commande de prothèses

Mot clés : electromyogramme, prothèses, décomposition d'EMG, simulation d'EMG

Résumé : Le pilotage moderne de prothèse robotisée de bras peut être sensiblement amélioré par l'utilisation de la décomposition d'EMG. Cette technique permet d'extraire l'activité des motoneurones de la moelle épinière, une représentation directe de la commande neuronale. Cette activité, qui est insensible aux facteurs non-liés au mouvement, tels que le type ou la position d'électrode EMG, est essentielle pour le pilotage des prothèses. Cependant les méthodes de décomposition existantes ne fournissent que l'activité d'un nombre limité de motoneurones. Cette information peut être considérée insuffisante pour en inférer l'intention de l'utilisateur. Dans ce travail, nous présentons une approche probabiliste qui utilise les modèles existants de la relation entre les activités des motoneurones et le mouvement. Nous comparons cette approche à une approche plus conventionnelle et montrons qu'elle fournit de meilleurs résultats même quand elle est alimentée avec un nombre très bas de motoneurones décomposés. Pour évaluer sa performance dans un environnement contrôlé, nous avons développé un modèle physiologique de simulation d'EMG et de contraction de muscle. De plus, une analyse sur les signaux expérimentaux a été réalisée.

Title: Probabilistic models based on EMG decomposition for prosthetic control

Keywords: electromyography, prosthetics, EMG decomposition, EMG simulation

Abstract: Modern prosthetic control can be significantly enhanced due to the use of EMG decomposition. This technique permits to extract the activity of motor neurons that control the movement, thus giving a direct representation of neural command. This activity, being unaltered by factors non-related to motion, such as type and position of EMG electrode, is of great interest in prosthetic control. Existing real-time decomposition methods, however, provide activities of a very limited number of motor neurons (up to ten). This can be considered insufficient for intent inference. In this work, we present a probabilistic approach to intent inference that uses existing models of relations between the behavior of motor neurons and the movement. We compare our approach with a conventional one presented in the literature and show that it produces significantly better results when provided with a small number of decomposed motor neurons. To assess its performance in a fully controlled environment, we have developed a physiology-based simulation model of EMG and muscle contraction. Moreover, the analysis was also performed using experimental recordings of muscle contractions.