Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in
Computer Engineering

Discrete and continuous control for brain-actuated robotic devices

Supervisor:
Prof. Luca Tonin
Co-supervisor:
Prof. Gloria Beraldo

Student:
Paolo Forin
ID number:
2019146

Academic year 2022/2023
Graduation date 17/04/2023

Thanks

I want to start by expressing my deep thanks to my thesis supervisors, Prof. Gloria Beraldo and Prof. Luca Tonin, for their unwavering support of my thesis research and study, as well as for their patience, inspiration, excitement, and vast knowledge. Additionally, my sincere gratitude to my friends and colleagues at the IAS-Lab, for giving me serenity and the desire to keep going.

I would also like to thank all the subjects who volunteered for this experiment; all this work would not have been possible without them.

A special dedication to my family, especially to my parents, brothers, and sisters. They have always supported and encouraged me during my studies. Also, a big thank you to those who spent their time praying that my exams went well.

A big hug and thanks to all my friends who have always been there for me through good and bad times.

Lastly, a huge thank you to all those people who inspired me and encouraged me to always push beyond the maximum.

Abstract

Brain-machine interface systems (BMIs) allow people to control external devices, such as powered wheelchairs, telepresence robots, robotic arms, exoskeletons, mouse and keyboards, through brain signals. A BMIs is defined as a closed-loop composed of acquisition, processing, feature extraction, classification and feedback. To interact with the devices there are two different control strategies to convert the BMI decoder's output into an appropriate signal for the robotic device.
The first one is called *discrete control* which allows the user to send discrete commands, therefore there are some seconds between two consecutive commands. In the case of BMIs based on motor imagery, for instance, a command is sent to the robot when a predetermined threshold is reached, or, in other words, when the control framework is certain of the user's intention.
The other is *continuous control*, which maps each brain signal to a control signal for the external device. Hence, continuous control increases precision and performance compared to discrete control, however, is more challenging to implement. A framework based on dynamical systems is a possible solution to implement such a continuous control. However, state-of-art control frameworks based on dynamical systems rely on several user-dependent hyper-parameters that make an optimal fine-tuning difficult for each user. In addition, the continuous control based on a dynamic system already proposed in the literature is designed to work symmetrically namely it implements the same behaviours for all the BMI classes. This constraint may be too strict, especially for a beginner user who has not yet mastered how to balance different classes in BMIs.
Therefore, the aim of the thesis is twofold: first, to reduce the number of parameters required for continuous control based on a dynamical system, and second, to allow the dynamical system to work with an asymmetrical behaviour, so each class can behave differently. To this end, we propose a new metric in order to find the most effective combination of parameters for each user.
In the first phase, we examine, a dataset available at IAS-Lab to evaluate the metric and find the proper correlation among parameters via a posterior analysis. Then, we recruited 12 subjects to perform a 2-class

motor imagery (MI) task by virtually controlling a steering wheel. The preliminary results confirm that there is a relation between the parameters used for the symmetrical and the asymmetrical dynamic system. Consequently, we validate a possible implementation of the asymmetrical dynamic system based on the relation found in the previous step with an experiment divided into three sessions. In these sessions, we compare the dynamic control system to the exponential one. The results of the experiment confirm that for almost all the users the asymmetric dynamical system provides better performance and less workload than the asymmetric exponential system.

Abstract

I sistemi di Brain-machine interface (BMIs) permettono alle persone di controllare apparecchiature esterne attraverso i segnali celebrali. Ad esempio, si possono controllare: sedie a rotelle, robot di telepresenza, bracci robotici, esoscheletri, mouse e tastiere. La BMI è definita da un loop chiuso composto da: acquisizione, elaborazione, estrazione delle feature, classificazione e feedback. Per interagire con le apparecchiature esterne ci sono due diverse strategie di controllo che convertono l'output del classificatore usato dalla BMI in un segnale appropriato per l'attrezzatura robotica.

Il primo si chiama *controllo discreto* il quale permette all'utente di inviare comandi discreti; quindi, che ci sono alcuni secondi che separano due comandi consecutivi. Nel caso di un sistema di BMI basato sulla motor imagery, per esempio, un comando viene inviato al robot quando una predefinita soglia viene raggiunta, o in altre parole, quando il framework di controllo è certo riguardo l'intenzione dell'utente.

L'altro è il *controllo continuo*, che mappa ogni segnale celebrale in un segnale di controllo per un device esterno. Quindi, il controllo continuo aumenta la precisione e le performance rispetto al controllo discreto, ma è più complicato da implementare. Un framework basato su un sistema dinamico è una possibile soluzione che implementa un controllo continuo. Però, lo stato dell'arte dei frameworks di controllo basati sui sistemi dinamici si basa su molti parametri che sono dipendenti all'utente, quindi questo rende difficile un'ottimizzazione di tali parametri per ogni soggetto. In più, il controllo continuo basato sul sistema dinamico già proposto nella letteratura lavora in modo simmetrico, quindi ha lo stesso comportamento per tutte le classi della BMI. Questa costrizione può essere troppo stretta specialmente per i nuovi utenti che non hanno ancora bilanciato le differenti classi usate nel sistema di BMI.

Dunque, questa tesi ha due scopi: il primo di ridurre il numero di parametri richiesti per il controllo continuo basato sul sistema dinamico, e il secondo, di permettere al sistema dinamico di lavorare in modo asimmetrico, quindi permettere alle classi di comportarsi diversamente. A questo scopo, noi proponiamo una nuova metrica per trovare la migliore combinazione di parametri per ogni utente.

Nella prima fase, noi esaminiamo un dataset disponibile presso IAS-Lab per valutare la metrica e trovare la correlazione tra i parametri attraverso questa analisi a posteriori. Poi, abbiamo reclutato 12 soggetti per eseguire un compito basato su 2-classi usando come paradigma la motor imagery (MI) permettendo il controllo virtuale di un volante.
I risultati preliminari confermano che c'è una relazione tra i parametri usati dal sistema dinamico sia per il caso simmetrico sia per quello asimmetrico. Di conseguenza, abbiamo validato una possibile implementazione del sistema dinamico asimmetrico basandoci sulla relazione trovata precedentemente con un esperimento diviso in tre sessioni. In queste sessioni abbiamo comparato il sistema dinamico con quello esponenziale. I risultati dell'esperimento confermano che per la maggior parte degli utenti il sistema dinamico asimmetrico fornisce migliori prestazioni e meno carico di lavoro rispetto al sistema esponenziale asimmetrico.

List of Figures

Contents

Chapter 1

Introduction

Many different disorders, such as amyotrophic lateral sclerosis (ALS), brainstem stroke, brain or spinal cord injury and multiple sclerosis can disrupt the neuromuscular channels through which the brain communicates with and control the external environment. The most studied disease by researchers is the one which causes the loss of control of all voluntary muscles and the person is locked in his/her body, unable to communicate in any way [1, 2].
There are three ways to restore the normal function of the body system [3]:

1. by increasing the capacity of the other pathways. For example, communicating using eye movements [4, 5],

2. by restoring the function and avoiding the break in the neural pathway which controls the muscle. For instance, people affected by spinal cord injury could use electromyographic (EMG) signals above the level of the lesion to allow the paralyzed muscle to move through electrical stimulation [6],

3. by using an alternative channel based on brain signals with a brain-machine interface (BMI) which allows the communication and control of external devices [3, 2].

1.1 Brain-machine interface (BMI)

The aim of brain-machine interface systems (BMIs) is to provide the brain with a new channel which allows the communication and the control of external devices such as powered wheelchairs, robotic arms, exoskeletons and telepresence robots [3, 2, 7, 1, 8].

Each BMIs relies on the so-called *closed-loop* (Figure 1.1) composed of several modules: acquisition, feature extraction, classification, control strategy and feedback [3, 8].

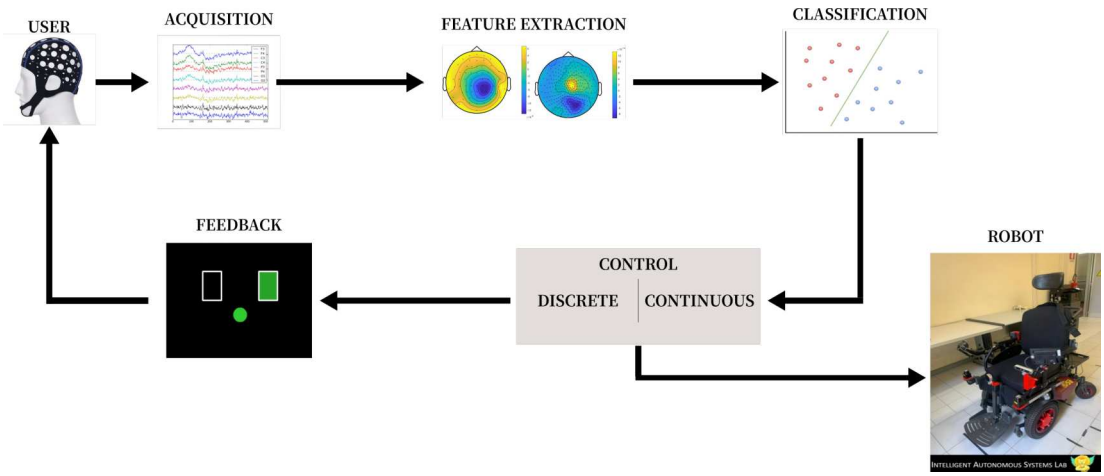


Figure 1.1: Closed-loop of a brain-machine interface (BMI).

- **The acquisition module.** It is in charge of recording brain activity. All BMI systems can be divided into two groups according to the scale at which the neural brain patterns are acquired [3, 7].

    1. *Invasive* BMIs can directly acquire a single neuron or a small group of them, they require surgery in order to plant some microelectrodes at the cortical and/or subcortical level. On the one hand, this methodology provides high-precision signals, on the other hand, a surgey is required and the user undergoes some medical risks, like electrodes infections [9, 10]
    2. *Non-invasive* BMIs can utilize hemodynamic responses (e.g., functional magnetic resonance imaging (fMRI) [11], functional near-infrared spectroscopy (NIRS) [12]) or the electrical activity in the brain (e.g., via electroencephalography (EEG) or magnetoencephalography (MEG)) [3, 8]. The most used method is EEG which allows high time resolution and does not need sophisticated instruments. Therefore, this method does not require surgeries, but the acquired brain signals have some noise.

    During the acquisition, the user is required to perform a specific mental task. There are two different paradigms: exogenous and endogenous stimulation.

In the first case, the signals depend on external stimulus (like light flashing), e.g., two systems that are well studied are P300 and Steady State Visually Evoked Potential (SSVEP) [13, 14]. In the second case, instead, the signal is the self-paced modulation of the brain rhythms of the user, and the most studied approach is the motor imagery (MI) that is based on the kinesthetic imagination of body parts [15, 16, 17, 18, 19, 20, 21, 22] (both paradigms will be further discussed in Section 1.2). Lastly, by combining different information signals we can have a hybrid BMI (hBMI), e.g., combining SSVEP with P300 [23].

- **The feature extraction module.** It filters the acquired signals with spatial and/or time filters in order to extract some features of interest in line with the required task. Based on these features, a detector is trained. For instance, we will use a Gaussian classifier and it is built up during the calibration phase. Then, the same classifier is used in real-time. Moreover, the output of the classifier a *raw probability* [16, 24, 25].

- **The control strategy.** It is in charge of converting the raw probability from the detector into a command for an external device. There are two different approaches in the literature [3]. The first one is the *discrete* strategy that allows sending discrete commands to the device with some seconds between two consecutive signals. For instance, in the system based on a BMI based on 2-class MI in [16], the commands are sent when the control framework is confident enough about the user's intention. The second is the *continuous* strategy which allows the system to continuously send the control signals to the robot. Both strategies will be further explained in Section 3.

- **The feedback.** It is a simple output that the system gives to the user in order to understand how the task is performed. It can be visual or auditory. For instance, in [16] the feedback for the MI task corresponds to two filling bars depending on the detected intent.

Every BMI system is based on a *closed-loop* architecture where the user and the machine interact with each other in order to achieve optimal control of the external device. A proven approach that allows users to improve their performances is mutual learning [26]. It claims that not only the machine learns from the user, but also the subject adapts his/her brain signals according to the provided feed-

back. This approach allows the creation of a stable system over time since it reduces the number of re-calibration of the decoder.

1.2 BMI Paradigms

A classification of *non-invasive* BMI systems is associated with different mental tasks the user is asked to perform in order to control the robotic application.

1.2.1 Exogenousm paradigms

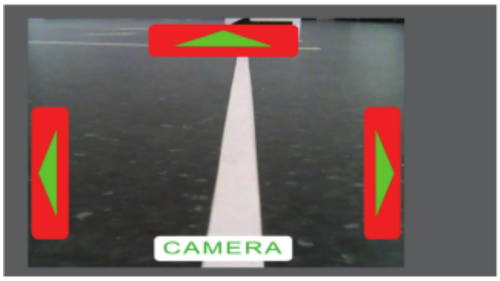
Exogenous paradigms relay on external stimuli in order to obtain a response in the users' EEG. For instance, exogenous BMI paradigms are Steady-State Evoked Potentials (SSVEPs) or P300 [2, 3]. The main limitation of this type of paradigm is that it depends on external stimuli.

Steady-State Evoked Potentials (SSVEPs). SSVEP is considered a continuous visual cortical response to constant frequency external stimuli on the retina. For instance, a flickering of a light-emitting diode (LED) can evoke an SSVEP. A typical SSVEP-based BMI system uses some LEDs to flicker with a different frequency, in order to achieve SSVEP associated with different external device actions [27]. Nowadays, the SSVEP is the most explored by the researchers among the exogenous paradigm [3], because:

- It allows to have a large number of BMI commands.
- The patterns are clearly distinguishable by the used frequencies. Since the stimulation frequency also corresponds to the user's response frequency.
- It does not require extensive training for the user.

Therefore, BMI systems based on SSVEP are able to decode the evoked response in the visual cortex elicited by a set of visual stimuli, each flickering at a given frequency [27]. The commands can be low-level (e.g., go forward, turn left, turn right, as in Figure 1.2a) or high-level (e.g., go in the kitchen, living room, bedroom as in Figure 1.2b) [28, 29].

P300. The evoked response in the EEG that is obtained after approximately 300ms due to the presence of stimulus in an oddball paradigm is called P300. In addition, a P300 signal is Event-Related Potential (ERP) since it measures



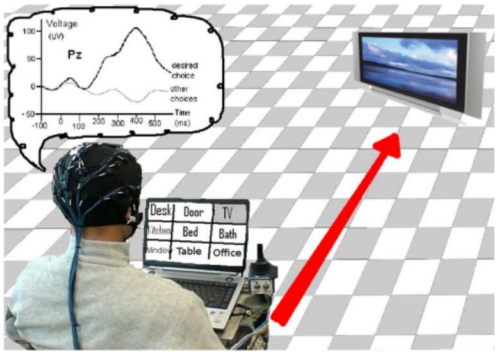
(a) The figure is taken from [28], it is the robot-GUI in drive mode in which the user must select the "go forward" command.

(b) The image is taken from [29], the user through SSVEP selects the final destination of the wheelchair.

Figure 1.2: Two different BMI systems based on SSVEP.

the brain's responses to a specific visual or auditory or somatosensory stimulus [27, 30]. In detail, a specific stimulus is a rare stimulus that the user must pay attention to over a sequence of stimuli.
The classical application of P300-based BMIs is for verbal communication [5, 31] (Figure 1.3a), but it can also be used to control external devices such as a powered wheelchair (Figure 1.3b) [14, 32, 33, 34].

(a) The user interface from which the user selects a letter after a column and then a row are intensified [5].



(b) The user selects the final destination by looking at the flashing final position in the computer (in this case the TV) [34].

Figure 1.3: Example of BMI application with P300.

1.2.2 Endogenous paradigms

Endogenous paradigm does not use external stimuli, it depends on the self-modulation of brain rhythms. The most studied approach by the researchers is motor imagery (MI).

Motor imagery (MI). In motor imagery (MI) users voluntarily modulate their sensorimotor rhythms by imagining the movement of a specific part of their body (e.g. right/left hand, both hands, or both feet. etc...) [3]. This voluntary modulation can be seen through the combination of event-related synchronization (ERS) and event-related desynchronization (ERD). Therefore, the phenomena reflecting sensorimotor brain activity regarding the decrease of amplitude is called ERD, instead, the increase of oscillatory component is called ERS [35]. In general, with MI paradigm, ERD and ERS are identified in the μ and β bands (8-14 Hz and 16-30 Hz) in the sensorimotor cortex. In addition, this modality requires information from an entire area, hence it works with a multichannel recording. Therefore exact locations and frequencies that the subject activates are user-dependent [3]. An example of ERD/ERS for a subject is reported in Figure 1.4.

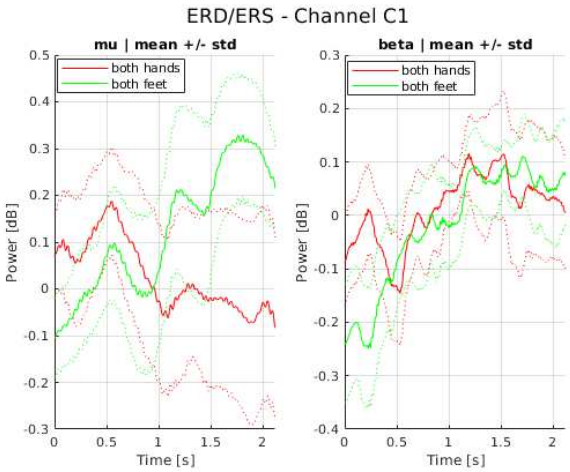
Brain-machine interface system based on MI, on the one hand, has high versatility since it does not depend on external events. In addition, this modality allows to deliver commands when the user voluntary executes a modulation in the EEG signal in order to perform the required mental task [36]. On the other hand, motor imagery requires continuous training in order to maximize the capacity of the user to activate the expected patterns. Other limitations of MI are the low number of degrees of freedom and the reduced number of actions that can be extracted from the neural modulations. In order to avoid these last two limitations often BMI systems based on MI are supported by shared intelligence [17, 37, 38]. For instance, BMI systems based on motor imagery can be used to drive external devices such as wheelchairs, robotic arms and teleoperation robots [15, 17, 20, 37, 39, 40].

1.3 BMI applications

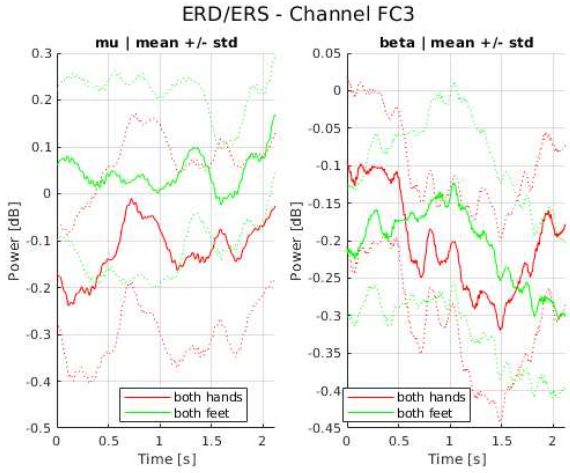
BMI system can be used for different applications, including communication and control, assistive mobility and entertainment [1, 2, 3, 8].

1.3.1 Communication and control

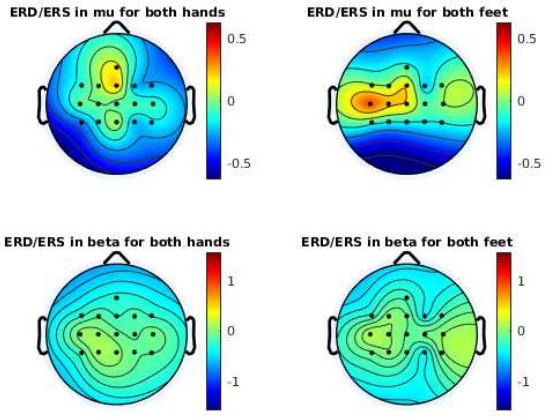
Brain-machine interface system allows users to communicate with the external environment. In [4, 5, 31] the user can select some symbols from the alphabet in order to build a word up. More into details, in [4] based on slow cortical potential (SCP) the user can move a mouse in a screen in order to select a group of letters. In the beginning, the alphabet is divided into two groups and the user must select



(a) The ERD/ERS of channel C1 is reported in the figure. Additionally, this channel is selected as features at 12 and 14 Hz. Thus, as we can see in the image, the μ band provides a good marker to differentiate between the signal for both hands or both feet, during a motor imagination task.



(b) In the figure, the ERD/ERS of channel FC3 that is not selected for the classifier is reported. Therefore, as we can see in the figure, there is not a huge difference between the two classes in both μ and β bands.



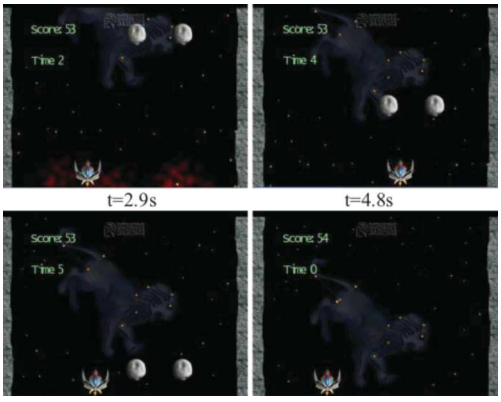
(c) In the figure the topoplot that shows the ERD/ERS is reported. As we can see, subject S8 is more able to modulate the μ band rather than the β band.

Figure 1.4: ERD/ERS of subject S8 for the same run is reported.

one of them subsequently, the selected group is divided into two new groups and the user must select one of them, this process goes on until a single letter is chosen. Instead in [5, 31] a BMIs based on P300 uses a grid of 6x6, containing all symbols from which the user can choose (Figure 1.3a). In this case, each row and each column, one at a time and randomly, are highlighted in order to solicit a P300 response. Then, a symbol is selected when the P300 signal for the rows and one for the columns are detected.
BMIs can be also used to control different types of external robots that act in the real world or in the virtual world. For example, in [19] a virtual helicopter is controlled with a 6-classes motor imagery protocol; then this work is extended in the real world [20]. On the other hand, in [15, 17, 28] telepresence robots must reach some target points or follow a predetermined path. In addition, the user must perform a motor imagery task in [15, 17], while in [28] a BMI system based on steady-state visual evoked potential is used.

1.3.2 Entertainment

BMI systems can be used, also, for entertainment. For instance, in [22] the user, looking at a screen, can drive a spaceship that should avoid collisions with some asteroids in Figure 1.5a. Moreover, in [41] many applications of the BMI systems for video games or virtual reality are reported as the one in Figure 1.5b. For instance, BMI based on SSVEP can be used to balance an animated 3D character that walks on a thread (Figure 1.5b). Another example reported in [41], used a motor imagery BMI system to make an avatar to "walk" in a virtual street and to stop in front of some avatars to talk with them.



(a) The game asteroids played with a BMI based on MI where continuous control feedback is provided to the user [22].

(b) A BMI system based on SSVEP is used in order to allow a 3D character to walk on a tightrope [41].

Figure 1.5: Example of games based on BMI.

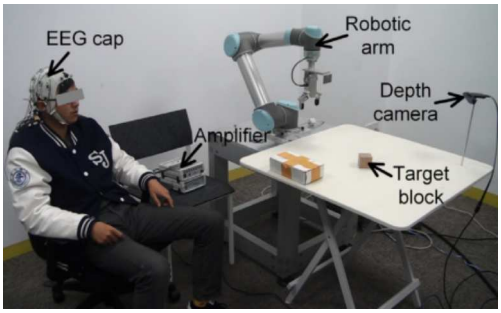
1.3.3 Assistive mobility

Brain-machine systems have the potential to assist paralyzed subjects by providing support with the movements, such as robotic arms or powered wheelchairs, as shown in Figure 1.6. For example, in [42] users can control external robots through BMI based on MI by selecting a specific action in a hierarchical menu. Moreover, they collect all the feasible actions in a central knowledge base that is continuously updated.
An aspect that influences all assistive mobility scenarios by improving the user's ability to use them, is the fusion of the user's commands and the robot's perception of the environment. This approach is known as shared control, shared autonomy and shared intelligence in literature. For instance, in [37] a BMI system based on MI provides commands, while the robot contextualizes them according to the current environment. Another example is [43], in which the user selects a high-level action, like reaching a place, and the robot does the action. An example that uses shared control with a robotic arm is [44] in which the robot plans and executes a path to reach an object detected with a camera, while the user provides commands, such us left front or right front, to help the robotic arm to reach as quickly as possible the detected object.

Robotic arms. BMIs can be used to control a robotic arm in order to grasp objects. For instance, in [44, 45] shared control allows the robotic arm to grasp an object. It is between the path planned by the robotic arm and the user that provide commands in order to help the system. In Figure 1.6a the set-up of the experiment [44] is reported. However, in [45] the user, in an initial phase, can also select the object to grasp via SSVEP stimulation. An example without the shared control is [46] where the user performs a MI task to select and grasp the object. A brain-machine interface can be also used to control a dual-arm robot, like in [18] where the user via BMIs based on MI makes the robot to lift and drop a box.

Powered wheelchair. Brain-machine systems can be used to support users' mobility. For instance, a BMI system based on P300 can be used to select the final high-level destination as in [34, 43], or in an unknown environment to choose an action or to pick as destination a calculated point near the actual position of the wheelchair like in [14, 32]. Furthermore, in [29] it is proved that a BMI system based on SSVEP can be used to control a powered wheelchair from a

command representing the final destination or the single action to perform, such as go forward. It is also possible to drive a wheelchair with a BMI based on MI. For instance, in [43] MI is used to select the final destination, while in [39] it is used to select an action that the wheelchair does, like go forward, turn left, turn right or go diagonally.



(a) A user with BMI based on MI grasps an object.

(b) The powered wheelchair used by the IAS-Lab of the University of Padua.

Figure 1.6: Two examples of assistive devices that can be controlled via BMI.

1.4 Brain-actuated robot

To use an external device with a BMI system, there are some aspects that must be emphasized. Firstly, BMI system provides a noisy channel and therefore there is a possibility of sending unintended commands, so it is important to contextualize them to avoid unexpected behaviors of the device. Secondly, the control system used to generate the control signal for the external device depends on the actions that the user must perform. All these aspects will be further described in the following subsections.

1.4.1 Intentional non-control

Most control system wants to maximize the accuracy and speed of the control signals. However, it must be taken into account when the user wants to send a command or when it does not want. These two states are well-known in literature as Intentional Control (IC) and Intentional non-control (INC) [47]. In the first, the user is performing the task, while in the second, the user is attentive and not intending to deliver commands even. However, it might happen that in the INC state some unintentional commands are sent anyway. To handle the INC state, the researchers explored two different approaches:

1. multi-class classification. Therefore, a class *REST* is included in the decoder classification. However, this solution increases the complexity of the system, since the resting class is unbounded [19, 20].

2. leaving the user to control the *unvolountary commands*. Thus, the subject actively controls the system in order to balance the classes. Nevertheless, this solution increases the workload for the user, because he/she needs to be always focused [15].

1.4.2 Discrete and continuous control system in the literature

As explained in Section 1.1, a control strategy converts the brain signals into commands for an external device. In the literature, there are two different approaches: discrete control system, and continuous control system [3].

1.4.2.1 Discrete control system

A discrete control system sends discrete commands. Therefore, two consecutive commands are separated by some seconds (e.g. 3s or 4s). This strategy improves the control signal's stability and reduces its variability. For these reasons, discrete control is the most applied for driving brain-actuated devices [3].
Looking at the paradigms that can be utilized with a brain-machine interface, we can notice that exogenous paradigms (like P300 and SSVEP) allow only discrete control systems since they depend on discrete external stimuli (e.g. flashing light at a different frequency). On the contrary, endogenous paradigms (e.g. MI) work with a discrete control system, although their nature since they depend on the voluntary modulation of brain rhythms. For instance, a BMI system based on MI

with a discrete control system has a continuous output of the decoder that can send a discrete command to the robot, only when a given threshold is reached [48], it will be explained in Section 3.1.

Discrete control systems have some limitations due to the period of time that the external device waits for a command, the presence of delay if a wrong command should be corrected and the low information transfer rate (0.3 commands per second on average) [3].

1.4.2.2 Continuous control system

A continuous control system sends commands to the external device continuously. Therefore each output of the BMI decoder is processed and it provides a control signal.
This strategy is more challenging to implement rather than the discrete, for two reasons, the first is due to the non-stationary nature of the EEG signal and the second is regarding the uncertainty of the BMI decoder output. However, continuous control increases the precision and the performance of the controlled device [15]. Due to its difficulty to be implemented, it is less used than the discrete approach and there are only few approaches in the literature:

1. The direct mapping of task-related neural signals into continuous driving commands. This approach uses modulated EEG signals that are less stable than classified ones. For instance, based on their previous work [19, 20] the user can control a quadcopter with 4-class motor imagery tasks in the real world.

2. The use of advanced control techniques to increase the reliability and stability of the probabilistic output of the BMI decoder. Therefore, a strong limitation is regarding the high number of required parameters. For example, in [22] the user controls a video game with a BMI based on 3-classes motor imagery, where the postprocessing chain is based on Savitzki–Golay filters, antibiasing strategy and multiple thresholding as explained also in [21].

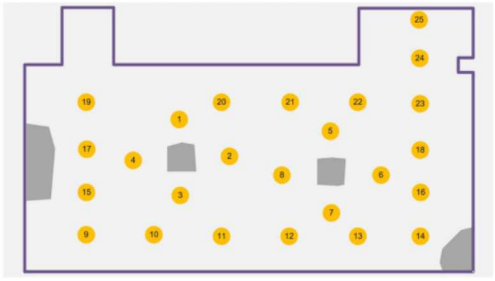
3. The use of a dynamic system, which takes into consideration the nature and the temporal evolution of the BMI output. It will be further explained in Section 3.2).

1.4.3 Shared control

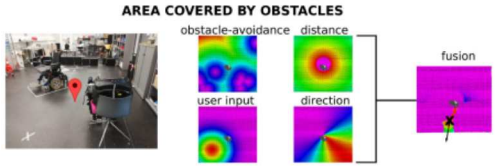
Non-invasive BMI systems allow a restricted number of commands that the user can deliver to the external device. To increase the number of commands and the stability of the whole system, two different approaches have been proposed:

1. The usage of the BMI to select the final target (as visible in Figure 1.7a) and the leaving of the planning and execution to the robot. The robot autonomously plans the best trajectory to reach the target. Typically, in this modality, the user cannot send any commands until the robot finishes the movement. For instance, in [14, 34, 42, 43] the user can select the final destination and the robot plans and follows the best path.

2. The merging of the BMI output with the robot perception. It allows the user to send high-level commands and it leaves all low-level commands to the robot. Therefore, each user command is interpreted and contextualised by the robot with the current surrounding environment. For instance, the user selects the direction of the robot and robot is in charge of avoiding obstacles [17, 37, 49], as shown in Figure 1.7b. In addition, several studies have proven that this approach improves the performance for healthy subjects and users suffering from motor disabilities [50, 51].



(a) The GUI map is constructed based on the obstacle map. The user selects the final position (yellow circle) with a BMI based on P300 (or MI) and the wheelchair follows the path in order to reach it avoiding the obstacles [43].



(b) Is an example of the fusion of policies (different grid probabilities maps) for the "area covered by obstacle" situation [17].

Figure 1.7: Two different approaches to add the robot intelligence in the closed-loop.

1.5 Motivations

The aim of this thesis is strictly correlated to the control system used in the BMI closed-loop (Figure 1.1). As highlighted in Section 1.4.2, there are two different

control systems that are available in the literature: the discrete and the continuous control system, further details will be explained in Section 3. In addition, for continuous control systems it is important that they allow INC status and give the user the feeling of being in control of the external device. Moreover, in order to ensure behaviors in line with the user's intentions, the creation of a control system must integrate some shared control techniques.
The continuous control system that will be analyzed is the one based on the dynamical system, that has higher precision and that the users feel it more natural, precise and easy to use than the classical discrete control, as proven in [15]. However, it has some limitations, such as the symmetrical behaviour that the F_{free} follows and the high number of parameters that must be tuned by the operator. The first limitation is a strict constraint, especially for naive users that have not mastered both classes.

1.5.1 Thesis aim

In this thesis, we investigate a continuous control system based on the dynamical system in order to allow asymmetrical behaviour and reduce the number of required parameters. The last aspect is important for the operator because it facilitates the tuning. Furthermore, we hypothesise that a BMI based on an asymmetric dynamic system achieves good results and users perceive more control, especially for naive users. Therefore, this thesis has a twofold aim: first, investigate if exists a relation among the parameters used for the dynamical system, and second, the implementation and validation of a two-classes MI BMI based on the asymmetric dynamical system using the relation founded at the first step.

Chapter 2

Methodology

This chapter describes the BMI implementation and the ROS-Neuro middleware exploited in the study.

2.1 The BMI based on motor imagery

To achieve a suitable classifier for a subject, several steps need to be performed. Firstly, we need to process the electroencephalography (EEG) data in order to remove noise, and then extract and select the most discriminating features for the two motor imagery classes. At this point, a classifier can be trained with the selected features and subsequently can be used by the BMI system.
Initially, the brain signal is filtered with two filters: a bandpass filter between 0.1 and 100 Hz, and a notch filter at 50 Hz. In our case, the filtering is directly computed by the EEG amplifier (g.USBAMP RESEARCH[1]). In order to increase the information of the filtered signals, a spatial filter is applied. We adopt a Laplacian filter that is a filter based on the second-order derivation of a single electrode and its neighbours. To create the feature vector based on the spectral power of the EEG signals, after the Laplacian filtering, we compute a power spectral density (PSD) through Welch's periodogram algorithm. Therefore, we use an overlapping window of 1 second (62.5 ms sliding) and we compute the PSD with 2 Hz resolution (from 4 to 48 Hz). For facilitating the creation of the classifier, the PSD is linearized with a base-ten logarithmic transformation.
At this point, a feature extraction and selection is made. Therefore, given the linearized PSD, we need to understand which frequencies and channels are more discriminative. With this aim, a Fisher Score is computed according to the for-

[1] `https://www.gtec.at/product/gusbamp-research/`

mula:

$$FS(c, f) = \frac{|\mu_1(c, f) - \mu_2(c, f)|}{\sqrt{\sigma_1^2(c, f) + \sigma_2^2(c, f)}} \tag{2.1}$$

where c, f are respectively a channel and a frequency, μ is the mean and σ is the standard deviation of that channel and frequency. Hence, we have a value for each channel at different frequencies; an example can be seen in Figures 2.1a and 2.1b. Then, these features can be used to train a Gaussian decoder, such as the one in Figure 2.1c, or as input for a classifier.

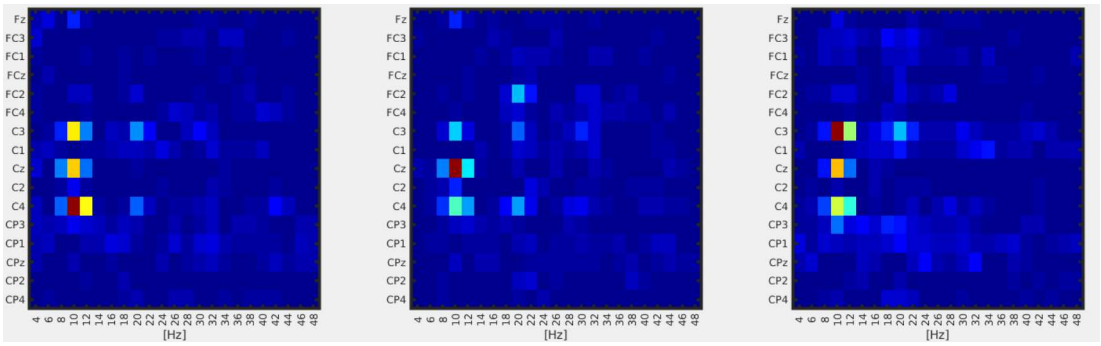
Then the most discriminative values are taken in order to train a Gaussian decoder as the one reported in Figure 2.1c.

2.2 Robot Operating System (ROS)

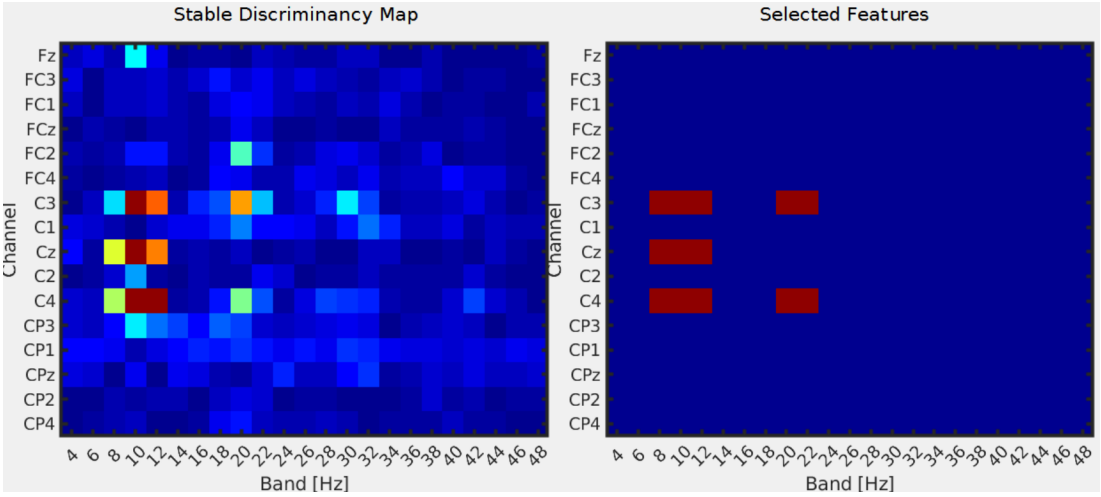
2.2.1 What is ROS?

Robot Operating System (ROS)[2] is an open-source framework used for developing robotics software. It provides a collection of tools, libraries, and conventions that are designed to simplify the process of creating complex and robust robot applications.
Starting from 2010, several distributions of ROS have been developed[3], the latest are: ROS Melodic Morenis and ROS Noetic Ninjemys that are suggested respectively for ubuntu 18.04 and 20.04; they were released on 2018 and 2020. More recently, ROS2[4] has been released, the main difference between ROS and ROS2 is related to the real-time since in ROS2 it is a primary aspect. In this thesis, ROS Melodic Moreins has been used.
ROS uses an architecture based on packages. With this structure, ROS encourages the developer to subdivide their code into packages that can be reused in other projects. Moreover, the code can be written in C++ or Python.
ROS architecture can be seen as a graph where each node is specialized to a function and each arch represents the communication between two nodes. In addition, in ROS there is a *ROS Master* node that allows peer-to-peer between different nodes.

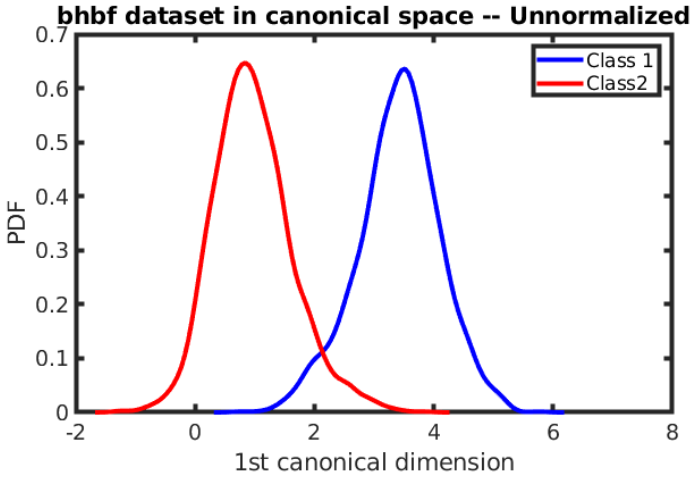
[2] `https://www.ros.org/`
[3] `http://wiki.ros.org/Distributions`
[4] `https://docs.ros.org/en/rolling/Releases.html`



(a) Features extraction with fisher score.



(b) Features selection.



(c) The Gaussian classifier of a subject with the features selected.

Figure 2.1: Features extraction, features selection and the Gaussian classifier a subject.

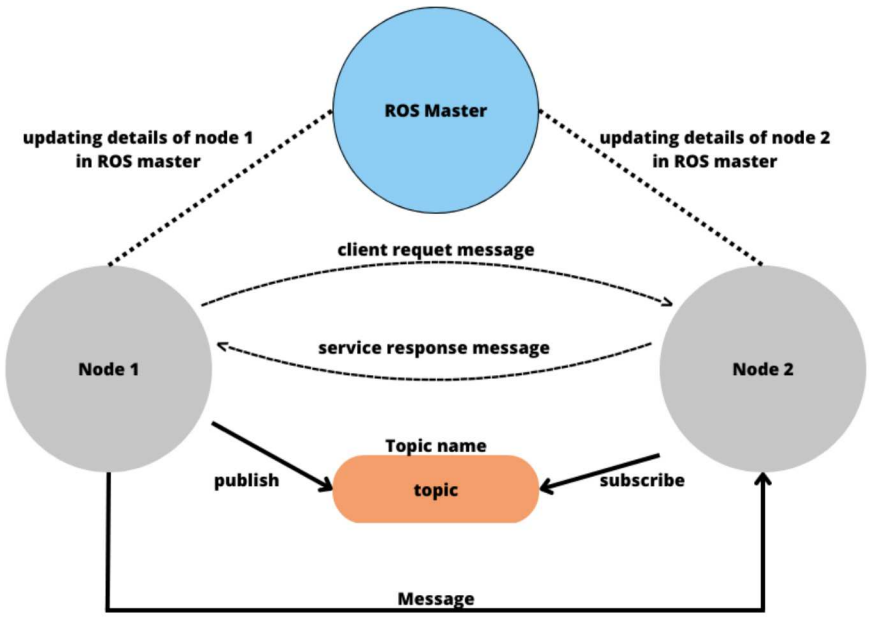
2.2.2 ROS communication

In ROS all the code is organized into packages, that contain one or more executables (nodes). The communication between nodes can be established in three different ways (an example is reported in Figure 2.2):

1. Message. It is a structure defined by the developer in a file saved with the extension *.msg* in a package. Therefore it is composed of a list of types, representing the field of the message (for instance: int, float, const, array, ...) and a name.

2. Topic. It allows asynchronous communication between different nodes. A topic is defined by a name and by the publication of only one type of message. In addition, a topic can have more publishers and more subscribers. Instead, a node can be subscribed to more topics. The idea behind the topic is a bus with a name, where each node can send or receive a message according to the publisher and subscriber division.

3. Service. It allows synchronous communication between two nodes with the request/reply strategy. One node is the client that through a *request message* sends a request to the service, that elaborates it and replies with a *response message*.



In the figure there is a representation of the three communications available on ROS. The first type is with a message that is sent from *Node 1* to *Node 2*. The second is through a topic, where *Node 1* publishes some messages (so it is a publisher) and *Node 2* read them (so it is a subscriber). The last is the service, where *Node 1* performs a request to *Node 2*, which elaborates it and replies with a response message.

Figure 2.2: ROS communication between two nodes

2.3 ROS-Neuro

ROS-Neuro[5] is an open-source neurorobotic middleware based on ROS (Figure 2.3). The initial concept of ROS-Neuro was introduced in [24]. The idea of ROS-Neuro is to achieve a common research framework to facilitate the expansion of the neurorobotics field. Authors have implemented all the packages for the acquisition, recording, processing, decision making and feedback for a 2-class motor imagery BMI [16, 25, 52].
Specifically, ROS-Neuro is composed of different packages in order to implement the closed-loop presented in Section 1.1:

- **rosneuro_acquisition**: it provides a ROS node that can communicate with some commercial amplifiers through the implementation of different plugins [25]. When the ROS node starts, the brain signals are acquired in chunks of data. The size of the chunks can be modified and is strictly related to the frame rate at which the node publishes a message in the topic */neurodata*. For instance, the chunk size is 32 if the data from the amplifier are acquired with a frequency of 512 Hz and the frame rate to publish them in the topic is 16 Hz.

- **rosneuro_recorder**: the related ROS node saves the acquired signals into BDF or GDF formats [25]. Once the node is launched, it performs a request with the */acquisition/get_info* service in order to set up the structure and to create the file in which the message published in the topic */neurodata* will be saved. In addition, it saves also events by reading the topic */events/bus* if the file format allows this type of information (GDF files allow this, instead BDFs do not).

- **rosneuro_processing**: this ROS node filters the data acquired in temporal, spectral and/or spatial domains [16]. It is subscribed to the topic */neurodata* in order to collect the EEG/EMG signals in a ring buffer, initially empty. When the buffer is full, the buffered data are processed. The first step is a spatial filter performed by a Laplacian matrix after computing a Power Spectral Density (PSD) using Welch's algorithm, and then an online classification is performed using a previously trained classifier. At this point the, so-called *raw probability* is published in the topic */smrbci/neuroprediction*. Some parameters like the buffer size, the laplacian matrix,

[5] `https://github.com/rosneuro`

the classifier and the parameters for Welch's algorithms can be set by the operator before the node starts.

- **rosneuro_decisionmaking**: it provides a ROS node that takes as input the *raw probability* read from the topic */smrbci/neuroprediction* and processes it in order to obtain a control signal [16]. It implements the control strategies described in Sections 3. This node publishes a message in the topic */events/bus* when a BMI command is been identified and the current value of the integrated probability in the topic */integrator/neuroprediction*. All the configuration parameters, like the thresholds, can be also set using ROS parameters. An important aspect to consider using this package is that the integrated probability must be reset to a uniform distribution when a command is recognized and published. Hence, the client node needs to perform it by using the service */integrator/reset*.

- **rosneuro_feedback**: it provides a node that, using OpenCV[6], shows a visual feedback [16]. This node displays some graphical elements, in a resizable window, according to the experimental protocol. It takes as input the message published in the topic */integrator/neuroprediction* in order to return the feedback through filling bars or moving the steering wheel. This package will be further explained in the Section 2.3.1.1.

In the current implementation of ROS-Neuro, the communication between different nodes is possible through a topic where a message is published by another node, or with a service where a specific action is asked by the client. For instance, the *decision-making* node is subscribed to the topic */smrbci/neuroprediction* where the *processing* node writes a *NeuroPrediction* message. As well, the *decision-making* node provides a service that gives the possibility to reset the control signal to an initial state. This service is useful when a task is finished and the control signal must be reset to an initial state.

2.3.1 A new feedback in the ROS-Neuro ecosystem

To use the dynamical system asymmetrically, we modify the current version of the decision-making in the ROS-Neuro repository. Furthermore, to simulate the control of a wheelchair we implement a digital wheel that turns according to our control system.

[6]https://opencv.org/

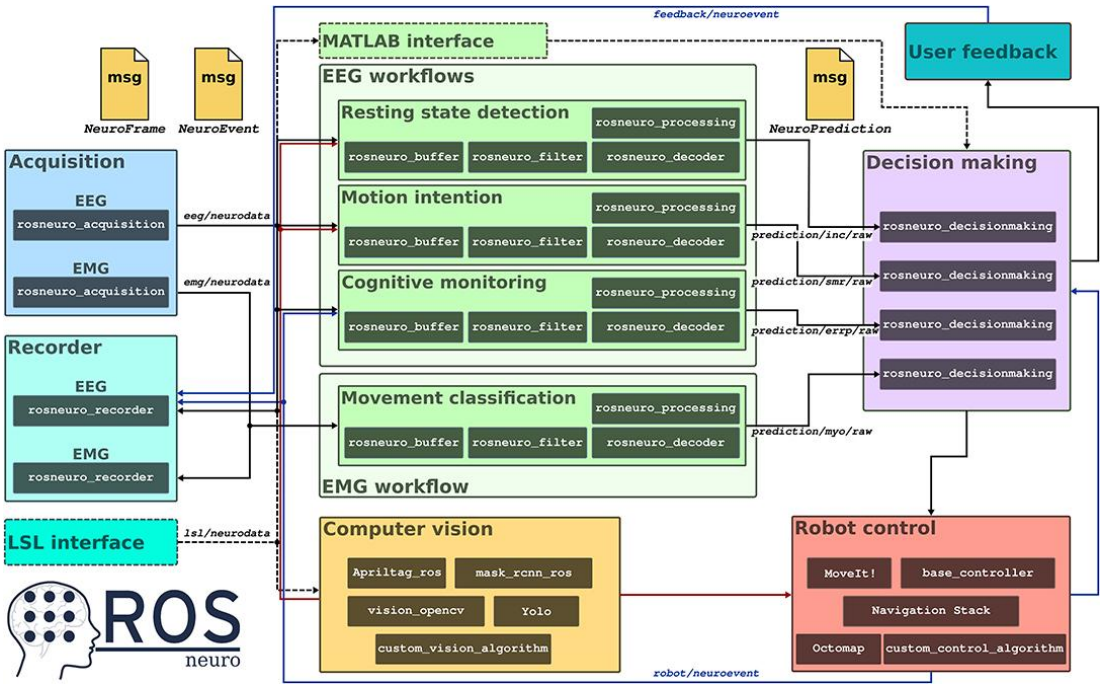


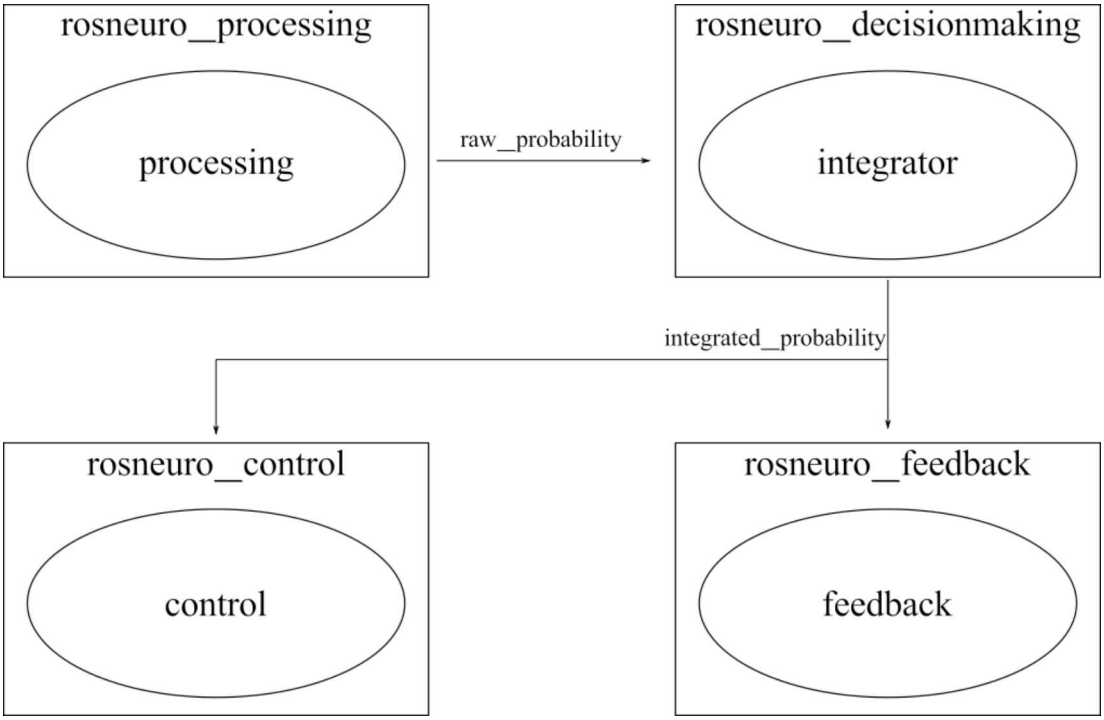
Figure 2.3: The figure reports the current version of ROS-Neuro available at `https://github.com/rosneuro`.

As shown in Figure 2.4, the *raw probability* is the probability as output from the processing node, that applies features selection and the decoder in order to have this initial probability. In addition, the integrated probability is the input for the feedback provided to the user and, it could also be the input to the external robotic device.

2.3.1.1 Feedback

In ROS-Neuro the feedback implemented is the one reported in Figure 2.5a. In this thesis, we implement a new version of the feedback shown in Figure 2.5b with the shape of a steering wheel.

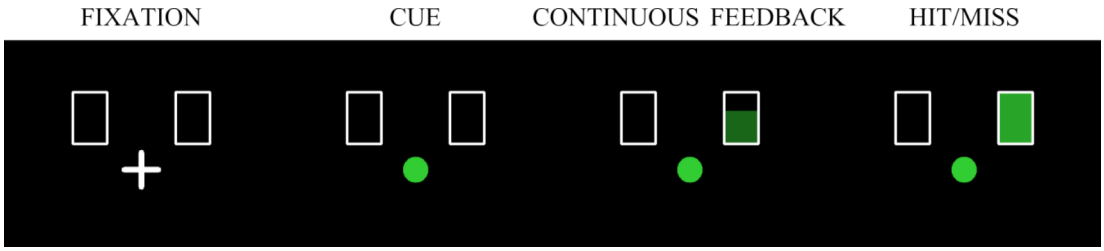
Every feedback must follow a sequence of periods, composed by fixation, cue, continuous feedback and hit/miss. During the *fixation*, the user can relax. In *cue*, the system shows the mental task to be performed by the subject. During *continuous feedback*, the user mentally controls the feedback. In *hit/miss* the subject reaches a predefined threshold. As seen in Figure 2.5, the sequence of actions that the wheel feedback performs is the same of the bar. The main difference, between the two kinds of feedback, is concerning how the control signal is mapped because, in the bar version, each class has its own bar (so we can have more than 2 classes), on the contrary, the wheel feedback allows only 2 classes



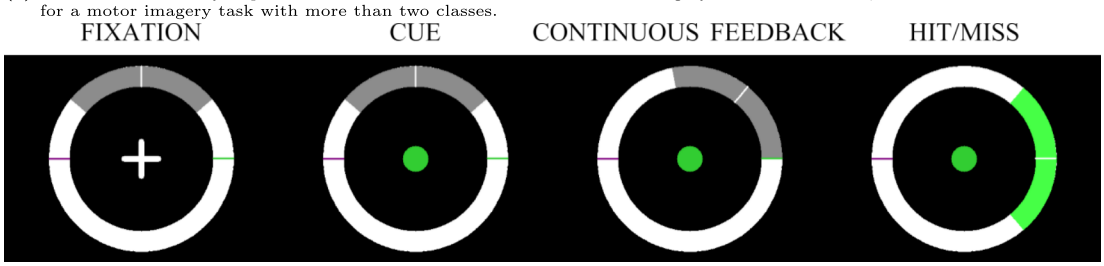
Four packages of ROS-Neuro. The processing package has as input the EEG signal, then it elaborates the brain signal and it applies to the elaborated signal a features selection and a decoder in order to obtain the raw probability. The decision-making package instead applies a control system, as explained in Section 3. The feedback node provides visual feedback to the user as explained in Section 2.3.1.1. At last, the control node allows the control of the external robotic device.

Figure 2.4: Some packages of ROS-Neuro

and the control signal is mapped from [0.0, 1.0] to [0°, 180°]. Thus, with only two classes we can use only one probability. For instance, if the raw probability vector is $[0.6, 0.4]$ we read only the first probability, then we translate it into the motion of the wheel.



(a) The feedback already implemented in ROS-Neuro for two class motor imagery tasks. In addition, this feedback can be used for a motor imagery task with more than two classes.



(b) The feedback that we implemented in order to simulate a steering wheel. Additionally, this feedback works only with two-classes motor imagery tasks.

In each sub-figure, the first image is regarding the fixation time. Then there is the cue period in which the user sees what MI task must perform. Then the continuous feedback, in which the user perceives his/her performance. Lastly, the hit period shows to the user if he/she reaches the threshold. In both cases, the user performs a motor imagery task of both feet.

Figure 2.5: The two different visual feedback.

Chapter 3

Evidence accumulation framework

To stabilize the signal output from the classifier and be more sure of the user's intention, an accumulation framework is needed. Therefore, the control system takes the *raw probability* output from the classifier and integrates it over time.

3.1 Exponential smoothing filter

BMI system based on MI provides a continuous output of the decoder. Therefore, to reduce system variability and increase signal stability, an integration over time is performed and compared with a predefined threshold. Typically, to do this, an exponential smoothing filter is used. It allows BMIs based on the decoding of voluntary modulations of brain rhythms to send discrete commands and it is defined by the following formula:

$$y_t = \alpha * x_t + (1 - \alpha) * y_{t-1} \tag{3.1}$$

where y_t is the current control signal, x_t is the posterior probability at time t (the output of the BMI decoder), y_{t-1} is the previously integrated control signal and α is the integration parameter. In addition, α controls the trade-off between the reliability of the BMI decoder output (x_t) and the importance of the previously integrated control signal (y_{t-1}). Moreover, with $t = 1$ the y_{t-1} value is set uniformly according to the number of classes (e.g., with BMI system based on MI with 2 classes $y_0 = 0.5$). Therefore, α is the only hyper-parameter that the operator must tune since it is strictly user-dependent. For instance, a control system with an high value of α trusts the classifier a lot, on the contrary if α is small. A signal is sent to the external device only when a predetermined threshold is

reached, or in other words only when the system is confident enough about the user's intent (e.g., $y_t > 0.7$) [48, 53].

3.2 Dynamical system

The dynamical system allows the generation of continuous commands for the external device. It is defined for a BMI based on 2-class motor imagery, as the combination of two forces:

$$\Delta y_t = \chi \cdot [\phi \cdot F_{free}(y_{t-1}) + (1 - \phi) \cdot F_{BMI}(x_t)] \tag{3.2}$$

where y_{t-1} is the previous control signal, x_t is the current BMI output, ϕ controls the contribution of the two forces, χ determines the velocity of the system. The last two parameters may be tuned by the operator, for instance, low ϕ means that there is a confident BMI decoder, instead high χ is related to a high reactiveness of the system. Therefore, the signal at time t is computed as:

$$y_t = y_{t-1} + \Delta y_t \tag{3.3}$$

F_{free} is designed in order to provide a conservative force when the control signal is close to 0.5 and a pushing force otherwise (see Figure 3.1a). Therefore, it is defined as:

$$F_{free}(y) = \begin{cases} -sin\left(\frac{\pi}{0.5-\omega} \cdot y\right) & if \quad y \in [0.0, 0.5 - \omega) \\ -\psi \cdot sin\left[\frac{\pi}{\omega} \cdot (y - 0.5)\right] & if \quad y \in [0.5 - \omega, 0.5 + \omega] \\ sin\left[\frac{\pi}{0.5-\omega} \cdot (y - 0.5 - \omega)\right] & if \quad y \in (0.5 + \omega, 1.0] \end{cases} \tag{3.4}$$

where $\psi > 0$ and y is the control signal. Therefore, the F_{free} divides the control signal space into many zones:

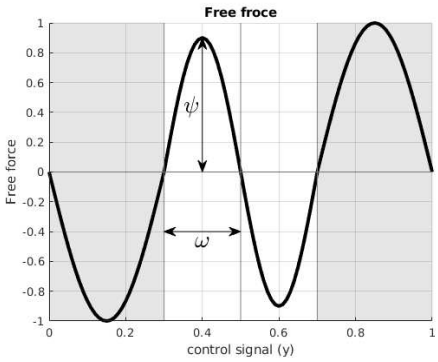
- $[0.0, 0.5 - \omega)$ and $(0.5 + \omega, 1.0]$. They are pushing zones. If the control signal is inside one of these zones, the F_{free} pushes to the high value for the respective class.

- $[0.5 - \omega, 0.5 + \omega]$. It is conservative zone. In this case, the F_{free} tries to avoid the sending of unwanted high control signals by keeping the control signal under the threshold.

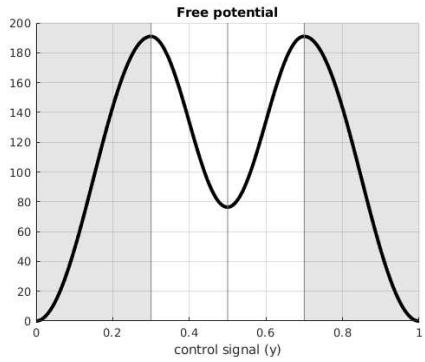
These subdivisions of the control signal space emphasise the different goal that the authors aimed for this new framework:

1. should manage the inconsistent BMI decoder output;
2. should help users when are actively performing the MI task to deliver the corresponding commands (IC state);
3. should limit the sending of undesired commands during resting state (INC state).

Given that, the user can be into two different states, the intentional control (IC) or the intentional noncontrol (INC), conservative zone is helpful when the user does not want to send any commands (INC state) but oscillations in the control signal can be arisen, instead, pushing zone is useful when the user wants to send a real command (IC state), so when he/she is inclined to perform the task properly.



(a) F_{free} with $\omega = 0.2$ and $\psi = 0.9$. The control signal space is divided into three spaces where a pushing force (grey zones) or a conservative force (white zone) is applied.



(b) Free force potential. It is derived from the F_{free} reported in a.

Figure 3.1: Design of the F_{free} and the free force potential.

In the Equation 3.2, the two forces have different aims. F_{BMI} perturbs the system according to the output of the BMI decoder. Instead, F_{BMI} is designed in order to strengthen the BMI decoder output with high confidence (close to -1.0 or 1.0) and to reduce the impact of uncertain decoded signals. The graphical representation of the shape of F_{BMI} is shown in Figure 3.2. Therefore it is defined by the following formula:

$$F_{BMI}(x) = 6.4 \cdot (x - 0.5)^3 + 0.4 \cdot (x - 0.5) \tag{3.5}$$

where x is the BMI decoder output.

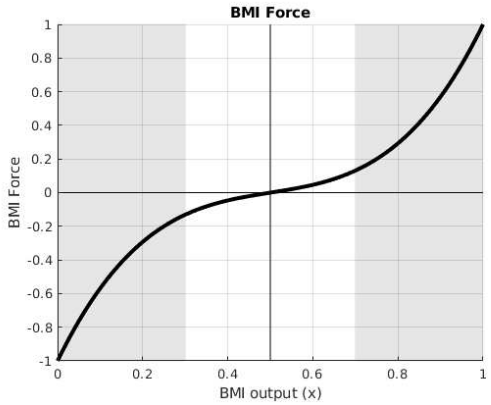


Figure 3.2: Graphical representation of the function used to generate the BMI force.

The dynamical system has a substantial limitation due to the current F_{free} shape that is rotational-symmetrical with respect to 0.5. This means that it has the same effect for the two classes [15]. Therefore, for a subject, especially for the naive one, it is a significant limitation because he/she can be strong with one class and weak with the other. In addition, with this control system, the operator must tune 4 hyper-parameters: ω, ψ, ϕ, χ for each subject since they are strictly user-dependent. For instance, a user can require a large conservative zone in order to stay more in INC, while another user can require a small one since he/she is good to remain at INC. Further details about the dynamic control system are in [15].

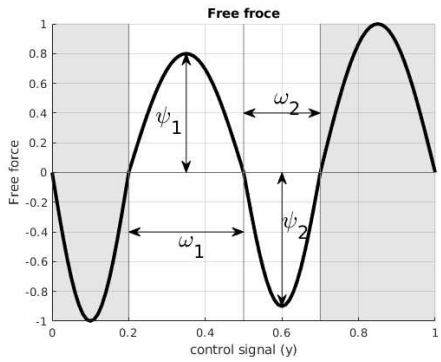
3.3 Modified dynamical system

In this thesis, we propose a modification of the F_{free} used in the dynamical system in order to allow asymmetrical behaviour for the two classes. The asymmetrical behaviour can be achieved by assigning different values of ω and ψ. Hence, the Equation 3.4 becomes:

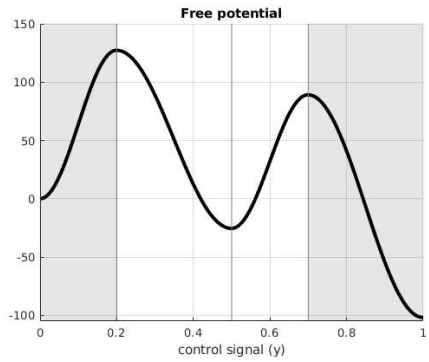
$$F_{free}(y) = \begin{cases} -sin\left(\frac{\pi}{0.5-\omega_1} \cdot y\right) & if \quad y \in [0.0, 0.5-\omega_1) \\ -\psi_1 \cdot sin\left[\frac{\pi}{\omega_1} \cdot (y-0.5)\right] & if \quad y \in [0.5-\omega_1, 0.5) \\ -\psi_2 \cdot sin\left[\frac{\pi}{\omega_2} \cdot (y-0.5)\right] & if \quad y \in [0.5, 0.5+\omega_2] \\ sin\left[\frac{\pi}{0.5-\omega_2} \cdot (y-0.5-\omega_2)\right] & if \quad y \in (0.5+\omega_2, 1.0] \end{cases} \tag{3.6}$$

A graphical representation of the updated F_{free} is shown in Figure 3.3a. With this new formulation of the dynamical system, we introduce new parameters to

manage the two classes differently, hence the operator must tune more parameters than before. To deal with this drawback, we have investigated if exists a relation among the parameters. Please refer to Section 3.3.1 for further details.



(a) F_{free} with $\omega_1 = 0.3, \omega_2 = 0.2$ and $\psi_1 = 0.8, \psi_2 = 0.9$. The control signal space is divided into three spaces: two pushing areas at the extreme and one conservative in the middle.



(b) Free force potential. It is derived from the F_{free} in a.

Figure 3.3: Design of the F_{free} and the free force potential with $\omega_1 \neq \omega_2$.

3.3.1 Relation between hyper-parameters

As anticipated before, to try to reduce the number of parameters, in this thesis, we verify if there is a relation among them. We have focused on the F_{free} since we hyphothesize a strong correlation between ω and ψ in the Equation 3.4:

- ω that defines the size of the conservative zone. With highest ω a stronger "resistance" is achieved;

- ψ that represents the "amount of resistance/help" to escape from the conservative zone. With the higher value of ψ, more difficult is to change the system state and to climb over the valley presented by the free force potential in the conservative zone (see Figure 3.3b).

Moreover, we fix the other parameters according to Table 3.1 to avoid introducing confounding factors.

To find the best values of the two parameters, ω and ψ, first, we conducted a data-driven optimization on a pre-collected dataset by optimising a metric that we design according to the criteria explained in Section 3.3.2. Then, we applied a regression analysis on the best-achieved values for each subject to find the relation between ω and ψ.

Parameters	Values
χ	1.00
ϕ	0.60
ω	from 0.025 to 0.475 with step of 0.025
ψ	from 0.05 to 1.0 with step of 0.05
th_{min}	0.45
th_{max}	0.55

Table 3.1: The values of the examined parameters, where χ, Φ, ω, ψ appear in the Equation 3.2 and 3.4, th_{min} and th_{max} are related to the Equation 3.12 used to optimise the parameters ω and ψ.

3.3.2 Evaluation of the relation between parameters

A dataset was used in order to find optimal values for a continuous control strategy based on the dynamical system described above. These data were stored at the Intelligent Autonomous System Laboratory (IAS-Lab)[1] at the University of Padua. The dataset is composed of eleventh different subjects (S1-S11) that perform a 2-class motor imagery task and it is recorded with ROS-Neuro[2] framework with a discrete control system based on the exponential smoothing filter [16]. Such data are composed of 520 trials divided into 260 per class (i.e., both hands vs. both feet). In addition, three subjects (S4, S5, S7) have no previous experience with BMI.
It is worth mentioning that this dataset is used to perform posterior analyses. Therefore, we can have only a general idea about the relation between the parameters used in the dynamical control system.

As previously explained we want to optimise the ω and ψ of the Equation 3.4. With this purpose, we have proposed a new metric taking into account the following aspects:

- We want the control signal to be upper or lower than the predefined band instead of a single threshold. Therefore, we want to limit the oscillation from up to bottom and vice-versa of the band.

- We want to maximize the time in which the control signal is upper or lower than the predefined band. Furthermore, we penalise the scoring of the control signal by attributing zero, when it belongs to the band.

[1] `http://robotics.dei.unipd.it/`
[2] `https://github.com/rosneuro`

- We want to minimize the time needed to send the command. Moreover, as demonstrated in the previous studies, it is infeasible that a control signal is sent before 1 s. Therefore, we minimize the time required to send a command and we filter the unfeasible values (< 1 s).

In addition, the control signal y is computed according to the Equation 3.3, and as explained in Section 3.2, the control signal depends on the combination of two forces: F_{free} and F_{BMI}. Moreover, the F_{free} depends on the parameters ω and ψ. Since all the other parameters are fixed as shown in Table 3.1, we can directly refer to the control signal as $y(\omega, \psi)$.
From these requirements, we have design a metric that for each combination of ω and ψ computes a score and it is based on the following three constraints:

1. we compute the signed distance between the current control signal y with the nearest extreme of the band $[th_{min},\ th_{max}]$ (Figure 3.4a), at each time $t \in [0, T_{trial}]$ with T_{trial} the duration of the trial. Therefore, we define the distant function $f_{distance_t}$ at time t as:

$$f_{distance_t}(y_t(\omega,\psi), th_{min}, th_{max}) = \begin{cases} 0 & if \ \ y_t(\omega,\psi) \in [th_{min}, th_{max}] \\ y_t(\omega,\psi) - th_{min} & if \ \ y_t(\omega,\psi) \in [0, th_{min}) \\ y_t(\omega,\psi) - th_{max} & if \ \ y_t(\omega,\psi) \in (th_{max}, 1] \end{cases} \tag{3.7}$$

2. we introduce a function to filter the infeasible commands. Therefore, we define $f_{discard_t}$, which depends on the selected couple $[\omega,\ \psi]$, in order to reject the control signals before 1 s (Figure 3.4b):

$$f_{discard_t}(\omega, \psi, th_{min}, th_{max}) = \begin{cases} 0 \quad if \ \ \exists (y_t(\omega,\psi) \leq th_{min} \vee y_t(\omega,\psi) \geq th_{max}) \\ \qquad with \ \ t \in [0,1]s \\ 1 \quad otherwise \end{cases} \tag{3.8}$$

3. we calculate the time that the control signal requires to reach the value 0 or 1 with the function f_{time} for each couple of candidates $[\omega,\ \psi]$. Therefore,

it is designed as:

$$f_{time}(\omega, \psi, th_{min}, th_{max}) = \begin{cases} t^* & if \ \ (y_{t^*}(\omega, \psi) = 0 \vee y_{t^*}(\omega, \psi) = 1) \\ & \wedge \ \ t^* > 1s \\ T_{trial} & otherwise \end{cases} \tag{3.9}$$

An example of the metric with simulated data is reported in Figure 3.4.

The final metric is a combination of the functions previously described. Initially, we combine the two Equations 3.7 and 3.8:

$$f_{final}(\omega, \psi) = \sum_{trial=1}^{Ntrials} (\sum_{t^*=0}^{T_{trial}} \sum_{t=0}^{t=t^*} f_{distance_t}(y_t(\omega, \psi), th_{min}, th_{max}) \cdot f_{discard_t}(\omega, \psi, th_{min}, th_{max})) \tag{3.10}$$

Then, we optimize the Equation 3.10 to search for the combinations of $[\omega, \psi]$ with the maximum score:

$$[\omega, \psi] = \max_{\omega, \psi} f_{final} \tag{3.11}$$

Finally, we choose the best ω^*, ψ^* among the candidates (obtained with the Equation3.11) using the following formula:

$$\omega^*, \psi^* = \min_{\omega_i, \psi_i \in [\omega, \psi]} (\sum_{tr=1}^{Ntrials} \sum_{t=0}^{T_{trial}} f_{time}(y_t(\omega_i, \psi_i), th_{min}, th_{max})) \tag{3.12}$$

Furthermore, since the dataset includes online runs previously recorded with the exponential discrete control system (explained in Section 3.1), for our pseudo-analysis we use as the ground-of-truth of each trial the predicted class.

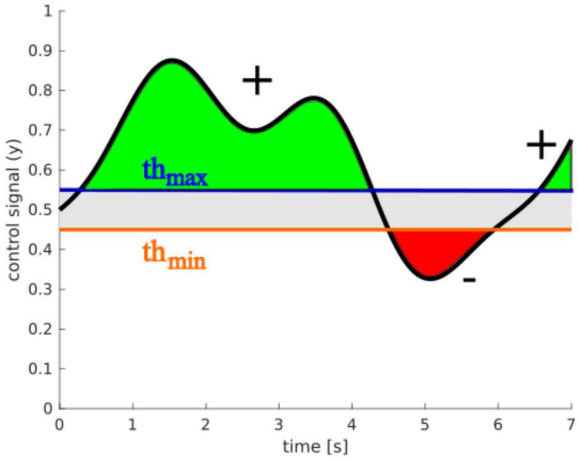
3.3.3 Preliminary results

The metric previously described identifies a single combination of ω and ψ. Therefore, the results are subdivided into two categories:

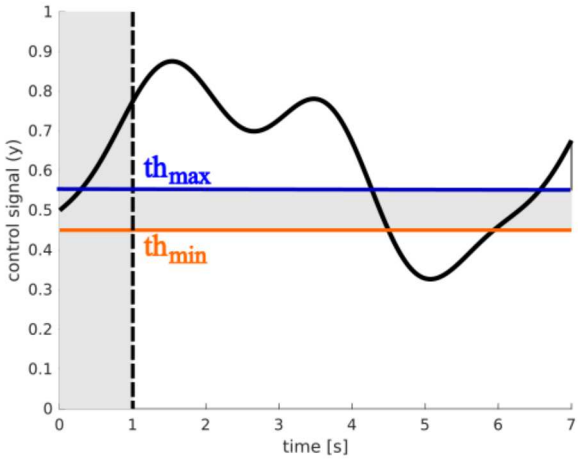
1. Symmetrical: search for a combination of ω and ψ that is the same for both classes.

2. Asymmetrical: search for a combination of ω and ψ that can be different for the two classes.

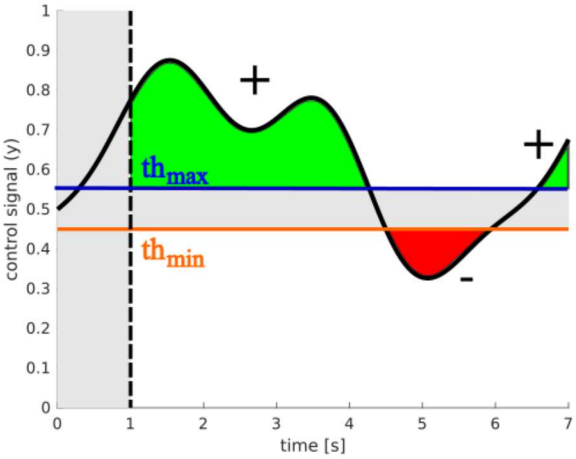
With these combinations of ω, ψ then a regression is applied to find a relation between ω and ψ. Therefore, we search for two different relations, one for the



(a) Contribution of the $f_{distance_t}$. Where in the general sum for these 7 seconds the green zone is associated with positive values, the red zone with negative values and the grey zone with null values.



(b) $f_{discard}$ function.



(c) The sum of the previous conditions.

Figure 3.4: Application of our metric with 7 seconds simulated trial and with $\omega = 0.2, \psi = 0.1$.

symmetrical case and the other for the asymmetrical one. The founded relations then need to be tested and validated with a following experiment, since we used prerecorded data to find them. Furthermore, in a real-time experiment, there is the influence of mutual learning between the system and the user.

3.3.3.1 Symmetrical case

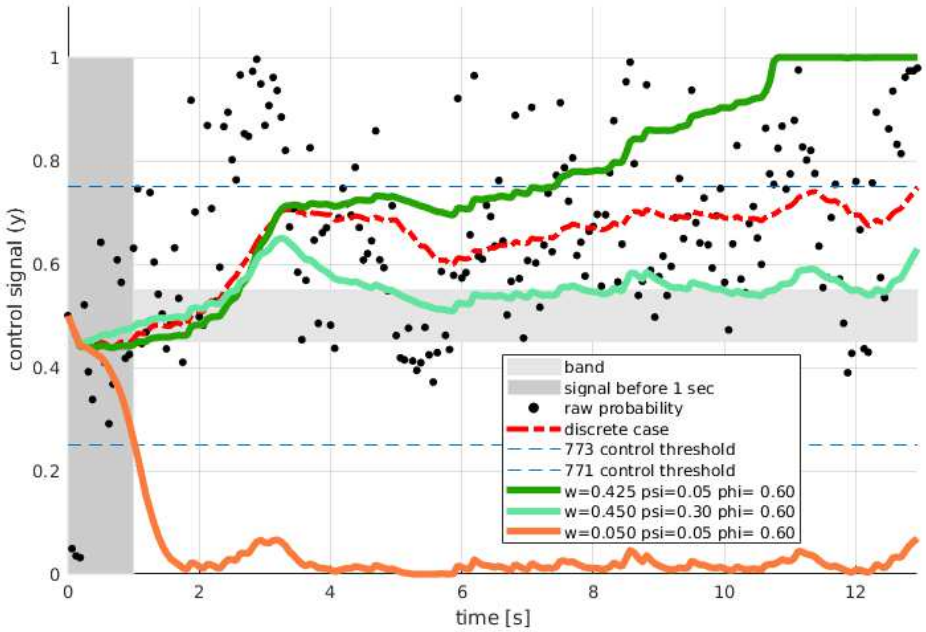
In the symmetrical case the metric explained in Section 3.3.2 provides the same value of ω and ψ for both classes. To clarify, Figure 3.5 shows an example of our metric for a trial where the user performs a both-hands task (upper case). We report the control signal associated with three combinations of ω and ψ, specifically, lines in green, cyan and orange are tied to the continuous control modality (dynamical system) and the red dashed is related to the discrete control modality (exponential smoothing system). In the figure, the best combination of ω and ψ, calculated by our metric, is in orange ($\omega = 0.2, \psi = 0.05$). The other combinations are discarded by the optimization. Table 3.2 reports the best ω and ψ for each subject. In addition, in the same table, there are also reported the accuracy associated with a discrete approach (exponential smoothing filter) and the accuracy of a continuous control system (dynamical system). Given these

Subject	ω	ψ	Accuracy discrete case	Accuracy continuous case, with discrete prediction as ground truth
S1	0.2	0.05	67.5%	96.25%
S2	0.025	1.00	72.5%	92.5%
S3	0.025	0.95	73.75%	85%
S4	0.475	0.05	92.5%	97.5%
S5	0.425	0.05	95%	90%
S6	0.025	1.00	92.5%	80%
S7	0.35	0.05	65%	76.67%
S8	0.175	0.20	66.67%	68.33%
S9	0.25	0.60	80%	83.33%
S10	0.4	0.05	81.11%	93.33%
S11	0.325	0.05	96.67%	91.67%

Comparison of the accuracy in the discrete case (e.g., via the exponential smoothing) vs. continuous case (e.g., via the optimised dynamical control framework).

Table 3.2: The best ω and ψ derived from the optimisation for each subject.

values, we apply a regression analysis in order to find a correlation between ω and



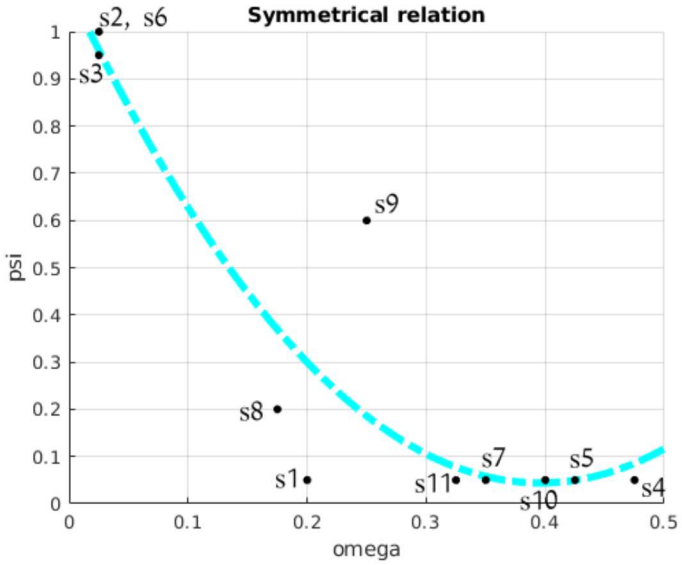
The signal is related to the subject S5 while he/she is performing a MI task of both hands (the control signal is 1 if the signal decode is both hands, and 0 if it is both feet). The figure reports in light grey the band selected according to the value of th_{min} and th_{max} in Table 3.1, in dark grey the Equation 3.8, in dashed light blue the control threshold used with the discrete control system, in bashed red the control signal compute with the exponential smoothing filter, in black dots the output of the classifier, and in green, light green, orange the control signal compute with the dynamical system. Moreover, the difference between the signal associated with green, orange and cyan is related to the different ω and ψ used. In detail, orange used $\omega = 0.05, \psi = 0.05$, green $\omega = 0.425, \psi = 0.05$ and light green $\omega = 0.45, \psi = 0.05$. The optimization returns as the best combination of ω and ψ the one used for the control signal reported in green. In this trial, the green is better than the light green because it reaches 1, and it is better than the orange because it follows better the raw probability (and also the classification is correct).

Figure 3.5: Example of our metric (symmetric behaviour) with real data.

ψ intending to verify our hypothesis. In Figure 3.6 we report the achieved results. We found a second-degree polynomial function $\psi = 6.6652 \cdot \omega^2 - 5.2772 \cdot \omega + 1.0884$. In addition, we use R^2 which measures the percentage of the dependent variable variation that our model can explain. Our metric reaches $R^2 = 0.8167$, which confirms that a relation between ω and ψ exists.

3.3.3.2 Asymmetrical case

In the asymmetrical case, it is possible to have different values between ωs and ψs per class. An example is reported in Figure 3.7, where the same subject (S5) and the same trial reported in Figure 3.5 are evaluated in the asymmetrical case. In this case, the optimal solution is the one in orange with $\omega_1 = 0.2$, $\omega_2 = 0.45$, $\psi_1 = 0.6$, $\psi_2 = 0.05$.



The black dots represent the optimal values for each subject. The curve in cyan represents the relation found from data and it is described with the equation: $\psi = 6.6652 \cdot \omega^2 - 5.2772 \cdot \omega + 1.0884$.

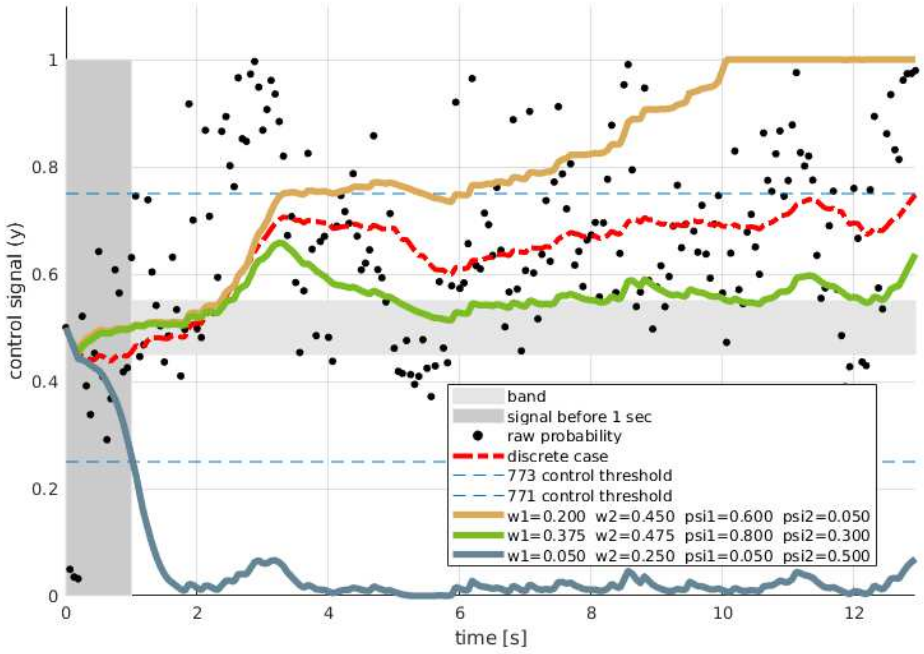
Figure 3.6: Dependency between ω and ψ in the symmetric case.

In Table 3.3 all the combinations for all eleven subjects are reported. In this case, our metric is used to find the best combination of $\omega_1,\ \omega_2,\ \psi_1,\ \psi_2$ without the constraint that $\omega_1 = \omega_2$. Furthermore, ranges for ω_1, ω_2 and ψ_1, ψ_2 are reported in Table 3.1.

Subject	ω_1	ω_2	ψ_1	ψ_2	Accuracy discrete case	Accuracy continuous case, with discrete prediction as ground truth
S1	0.275	0.475	0.10	0.05	67.5%	96.25%
S2	0.025	0.025	1.00	0.95	72.5%	87.5%
S3	0.025	0.025	0.85	1.00	73.75%	83.75%
S4	0.35	0.05	0.50	0.05	92.5%	95%
S5	0.20	0.45	0.60	0.05	95%	97.5%
S6	0.025	0.025	1.00	1.00	92.5%	80%
S7	0.375	0.10	0.30	0.10	65%	76.67%
S8	0.425	0.025	0.50	0.90	66.67%	73.33%
S9	0.45	0.25	0.05	0.95	80%	95%
S10	0.45	0.40	0.10	0.05	81.11%	92.22%
S11	0.075	0.275	0.05	0.05	96.67%	91.67%

Comparison of the accuracy in the discrete case (e.g., via the exponential smoothing) vs. continuous case (e.g., via the optimised dynamical control framework).

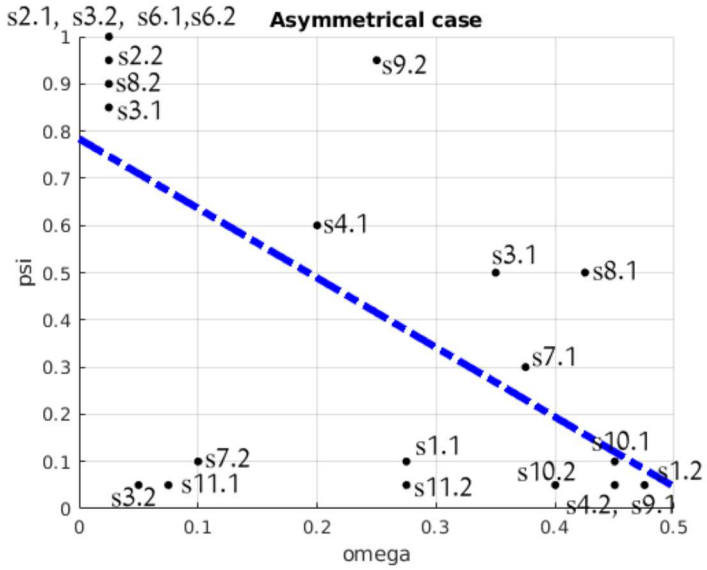
Table 3.3: The best $\omega_1,\ \omega_2$ and $\psi_1,\ \psi_2$ derived from the optimisation for each subject.



The signal is related to the subject S5 while he/she is performing a MI task of both hands (the control signal is 1 if the signal decode is both hands, and 0 if it is both feet). The figure reports in light grey the band selected according to the value of th_{min} and th_{max} in table 3.1, in dark grey the function 3.8, dashed light blue the control threshold used with the discrete control system, in bashed red the control signal compute with the exponential smoothing filter, in black dots the output of the classifier, and in green, brown, blue the control signal compute with the dynamical system. Moreover, the difference between the signal associated with green, brown and blue is related to the different ω and ψ used. In detail, brown used $\omega_1 = 0.2$, $\omega_2 = 0.45$, $\psi_1 = 0.60$, $\psi_2 = 0.05$, green $\omega_1 = 0.375$, $\omega_2 = 0.475$, $\psi_1 = 0.80$, $\psi_2 = 0.3$ and blue $\omega_1 = 0.05$, $\omega_2 = 0.25$, $\psi_1 = 0.05$, $\psi_2 = 0.5$. The optimization returns that the best combination of ωs and ψs is the one used for the control signal reported in brown. In this trial, the brown is better than the green because it reaches 1, and it is better than the blue because it follows better the raw probability (and also the classification is correct).

Figure 3.7: Example of our metric (asymmetric behaviour) with real data.

Given these values, we apply a regression analysis in order to find a correlation between ω and ψ. In this case, for each subject, we have two ω and two ψ. Therefore, the points plotted in Figure 3.8 are divided into the two classes. In this case, we found a linear relation $\psi = -1.4742 \cdot \omega + 0.7836$ with $R^2 = 0.4078$. We selected this relation since it has a higher R^2 value and since it is more manageable than a cubic or quadratic one as represented in Figure 3.8.



The black dots represent the optimal values of ω and ψ for each subject, divided into two classes: both hands ($< subject >.1$) and both feet ($< subject >.2$). The curve in blue represents the relation found from data and it is described with the equation: $\psi = -1.4742 \cdot \omega + 0.7836$.

Figure 3.8: Dependency between ω and ψ in the symmetric case.

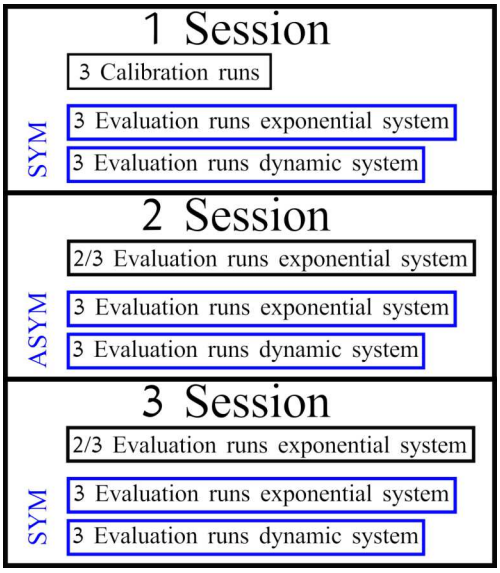
Chapter 4

Experimental design

The users are required to perform MI tasks of both hands versus both feet. Since the dynamic system provides better results for the INC state (at least for the symmetrical case as demonstrated in [15]) than the exponential, we add the *rest* task in the evaluation, in which the user must relax without thinking about hands or feet. In addition, users signed informed consent after the experiment was explained to them. Furthermore, the experiments were conducted in accordance with the Declaration of Helsinki.
Remembering the hypothesis and the aim of this thesis exposed in Section 1.5.1, we want to validate the relations between the parameters used in the dynamic system that we introduced in Section 3.3.3 through this experiment. In addition, we compare the dynamic system with the exponential one.

4.1 Protocol

The sessions of the experiment were distributed over three days in one month and the duration of each of them is about two hours and a half. Figure 4.1 schematically reports the pipeline followed in the experiment. The first session was dedicated to the calibration, with three runs, and to the evaluation with six shuffled runs. The aim of the evaluation runs was to compare the exponential and the dynamic system in the symmetrical modality. This means that for the exponential the thresholds that the user must reach are the same for both classes and for the dynamic that the same ω is chosen for both classes. Additionally, we fixed the thresholds to 0.7 and ω to 0.2, in this way we have the peaks of the potential free force that coincide with the thresholds used with the exponential system (Figure 3.1b). However, it is already proven that the dynamic system



The experiment is split into three days. On the first day, with the calibration runs, we create the decoder that can be updated in the following days by checking with the first two or three runs if the user has control of the feedback. Moreover, in the first session, we test the relation for the dynamic parameters reported in Figure 3.6. In the second session, we use the asymmetrical relation shown in Figure 3.8 for the dynamic system; instead for the exponential one we use thresholds with different values for both classes. Lastly, in the third session, we use the symmetrical relation between ω and ψ with the asymmetrical dynamic system and we compare it with the asymmetric exponential.

Figure 4.1: Schematic representation of the experimental structure.

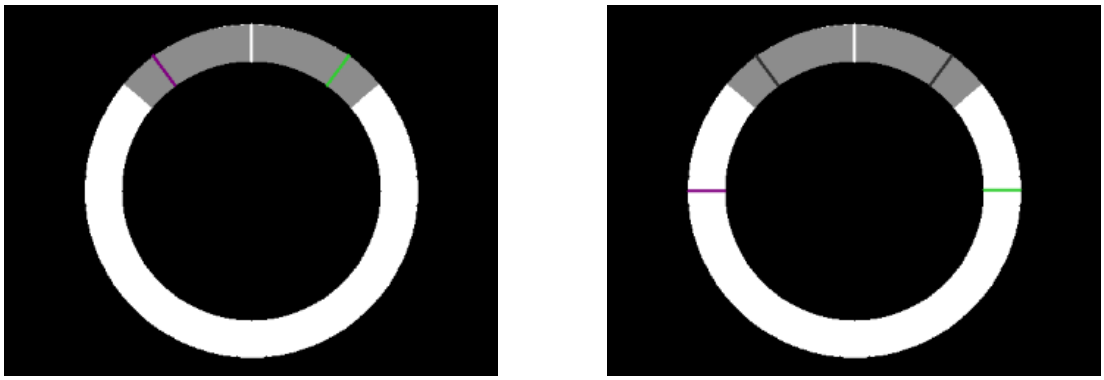
provides better results than the exponential with a symmetrical behaviour [15]. Nevertheless, we want to validate the founded relationship between ω and ψ (Figure 3.6), so we use it to calculate the value of ψ with $\omega = 0.2$.

The second session starts with two or three evaluation runs with the exponential system with the same threshold used in the first session, in order to check if is necessary to update the decoder because the user activates different channels than the ones selected in the first session. After this preliminary step, the user performs six evaluation runs, divided into three for the asymmetrical dynamic system and three for the asymmetrical exponential system. In this case, we manually select the thresholds used with the exponential system considering that the two thresholds must have different values. Instead, the values for the dynamical system are chosen considering the same transformation used in the first session, so: $\omega_1 = th_1 - 0.5$ and $\omega_2 = th_2 - 0.5$, subsequently, the values of ψ_1, ψ_2 are calculated with the relation founded for the asymmetrical case shown in Figure 3.8.

In the third session, as in the second one, the first two or three evaluation runs use the exponential system so that we can understand if it is necessary to update the decoder. Then three evaluation runs with an asymmetrical behaviour of the

exponential system and three with the dynamic asymmetric system are shuffled and tested. In addition, the relation used to calculate the relative ψ_1, ψ_2 for the dynamic system is the one in Figure 3.6. Moreover, the operator chooses the thresholds for the exponential system and the value of ω_1 and ω_2 for the dynamic system. Therefore, there are two significant differences between this session and the second one. Firstly, the relation used to calculate ψ_1 and ψ_2 is the one used also for the symmetrical case, shown in Figure 3.6. Secondly, the operator chooses manually the values for ω_1 and ω_2, like the threshold used for the exponential case.

In all the sessions, the evaluation runs that test the dynamic or the exponential system are shuffled, in this way, there is no learning effect. Additionally, each run is composed of 30 trials: 10 for both hands, 10 for both feet and 10 for rest (5 fixed feedback, which provide no feedback to the user, and 5 moving feedback, where the user moves the wheel). Since in the experiments we use the exponential and the dynamic control system, we add to the feedback some lines, as we can see in Figure 4.2. In this way, the user can understand which control system is been used. This is useful for the questionnaire administrated to the users at the end of each session.



(a) The steering wheel feedback with the exponential control system. In this feedback, only the thresholds that the user must reach with his/her brain signals are shown.

(b) The steering wheel feedback with the dynamic control system. This feedback shows the thresholds that the user must reach and two more black lines that represent the pick of the potential free force (explained in Section 3.2).

Figure 4.2: The two different visual feedback with the steering wheel.

For instance, Figure 4.3 shows a user that is performing the first session. Moreover, Figure 4.3a is a calibration run, where the user performs *both hands* task; Figure 4.3b is an evaluation run of the exponential control system where the user performs a *both feet* task; instead Figure 4.3c is an evaluation run of the dynamic control system where the user performs a *rest* task. Therefore we associate each colour with a motor imagery task: purple means both hands, green

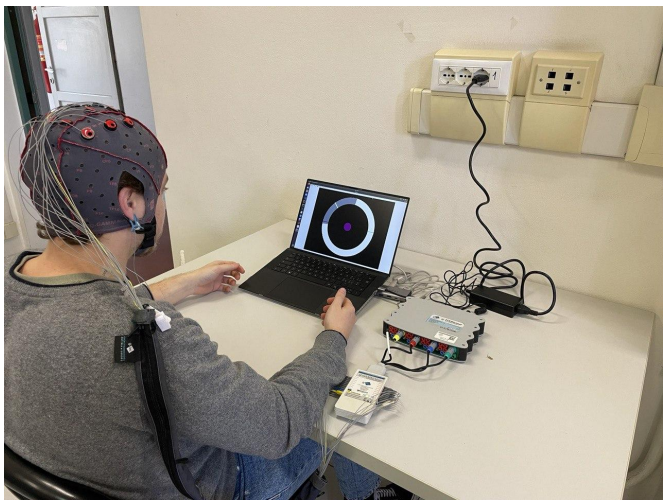
means both feet and blue means rest.

4.2 Users

For this study, 12 voluntary subjects take place in the experiment. All users were healthy and not affected by any pathologies. Additionally, the average age is 24 $\pm$ 3 years and all the users except for S12, have no experience with a BMI system. The participants dedicate three days in the month for the experiment, unfortunately, not all twelve have this possibility. Therefore, we allow some subjects to take part only in two of the three sessions in order to better validate our new possible outcome hypothesis between two different sessions.
Users were informed of the session's goals and the process for recording data before each session began. Furthermore, participants at each session, sign a consent document that has a full explanation of how recorded data will be used and how privacy will be treated.
Tables 4.1, 4.2 and 4.3 report the parameters used for the exponential control system and the dynamic one for each section and each subject.
Lastly, these users are not the same involved in the data collection used for the searching of the relation between ω and ψ reported in Section 3.3.2.

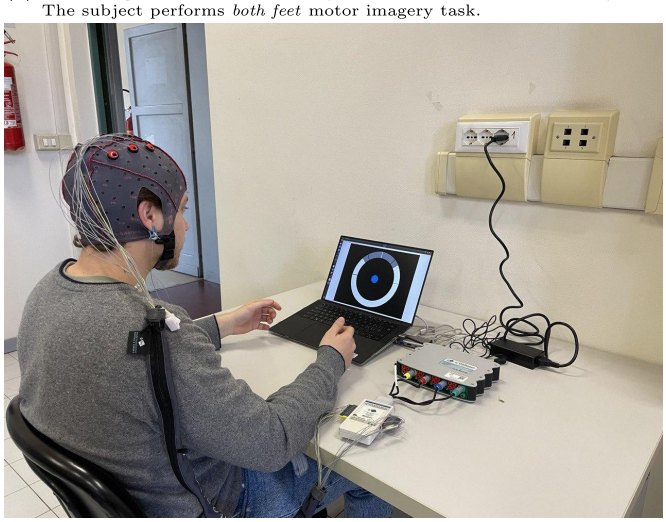
	Session 1	
Subject	Exponential parameters	Dynamic parameters
S1	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S2	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S3	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S4	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S5	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S6	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S7	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S8	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S9	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S10	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S11	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$
S12	bh=0.7, bf=0.7	$\omega = 0.20,\ \phi = 0.60$

Table 4.1: The table reports all the values used for the exponential and dynamical systems in the first session for each subject.



(a) User doing the calibration. The subject performs *both hands* motor imagery task.

(b) User doing the evaluation of the symmetrical exponential control system. The subject performs *both feet* motor imagery task.



(c) User doing the evaluation of the symmetrical dynamic control system. The subject performs *rest* motor imagery task.

Figure 4.3: A user performing the first session.

Subject	Session 2	
	Exponential parameters	Dynamic parameters
S1	bh=0.65, bf=0.70	$\omega_{bh} = 0.15,\ \omega_{bf} = 0.20,\ \phi = 0.60$
S2	bh=0.65, bf=0.70	$\omega_{bh} = 0.15,\ \omega_{bf} = 0.20,\ \phi = 0.60$
S3	bh=0.75, bf=0.85	$\omega_{bh} = 0.25,\ \omega_{bf} = 0.35,\ \phi = 0.60$
S4	bh=0.65, bf=0.70	$\omega_{bh} = 0.15,\ \omega_{bf} = 0.20,\ \phi = 0.60$
S5	bh=0.70, bf=0.75	$\omega_{bh} = 0.20,\ \omega_{bf} = 0.25,\ \phi = 0.60$
S6	bh=0.65, bf=0.75	$\omega_{bh} = 0.15,\ \omega_{bf} = 0.25,\ \phi = 0.60$
S7	bh=0.75, bf=0.65	$\omega_{bh} = 0.25,\ \omega_{bf} = 0.15,\ \phi = 0.60$
S8	bh=0.65, bf=0.70	$\omega_{bh} = 0.15,\ \omega_{bf} = 0.20,\ \phi = 0.60$
S9	-	-
S10	-	-
S11	-	-
S12	-	-

Table 4.2: The table reports all the values used for the exponential and dynamical systems in the second session for each subject. Additionally, some subjects have no value since they did not participate in this session.

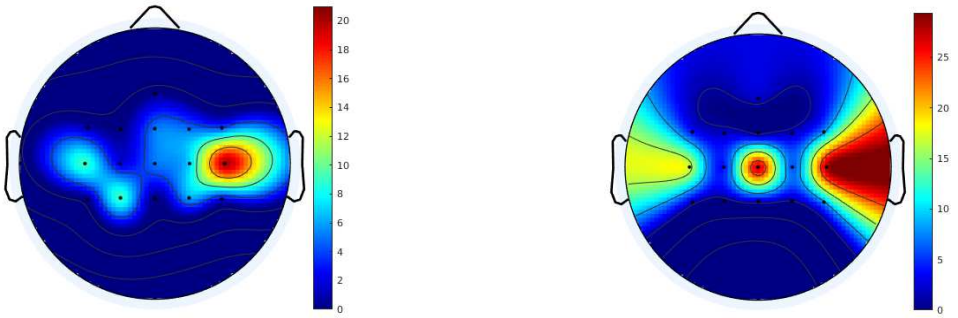
Subject	Session 3	
	Exponential parameters	Dynamic parameters
S1	bh=0.65, bf=0.70	$\omega_{bh} = 0.325,\ \omega_{bf} = 0.250,\ \phi = 0.40$
S2	bh=0.70, bf=0.65	$\omega_{bh} = 0.280,\ \omega_{bf} = 0.315,\ \phi = 0.46$
S3	bh=0.80, bf=0.75	$\omega_{bh} = 0.100,\ \omega_{bf} = 0.150,\ \phi = 0.60$
S4	bh=0.70, bf=0.60	$\omega_{bh} = 0.300,\ \omega_{bf} = 0.400,\ \phi = 0.40$
S5	-	-
S6	-	-
S7	-	-
S8	bh=0.65, bf=0.70	$\omega_{bh} = 0.315,\ \omega_{bf} = 0.275,\ \phi = 0.59$
S9	bh=0.80, bf=0.85	$\omega_{bh} = 0.215,\ \omega_{bf} = 0.190,\ \phi = 0.60$
S10	bh=0.65, bf=0.70	$\omega_{bh} = 0.380,\ \omega_{bf} = 0.330,\ \phi = 0.59$
S11	bh=0.70, bf=0.75	$\omega_{bh} = 0.400,\ \omega_{bf} = 0.300,\ \phi = 0.60$
S12	bh=0.65, bf=0.75	$\omega_{bh} = 0.400,\ \omega_{bf} = 0.150,\ \phi = 0.60$

Table 4.3: The table reports all the values used for the exponential and dynamical systems in the third session for each subject. Additionally, some subjects have no value since they did not participate in this session.

Chapter 5

Results

This chapter exposes all the results obtained in the three sessions. The first analysis is regarding the features selected for the decoder. Figure 5.1 shows the spatial and spectral consistency of the features exploited in the BMI decoder across participants. The most frequent activated channel is C4 in the μ band (20 times respectively versus 5 times for Cz), instead in the β band are Cz and C4 (27 and 28 times respectively).



(a) Most selected features for μ bands.

(b) Most selected features for β bands..

Figure 5.1: Topographic representation of the most selected features during the calibration phase for μ and β bands.

Then, for each subject and for each session we analyze:

- general accuracy. It is expressed as:

$$acc_s = \frac{Number\ of\ hit\ correct}{Number\ of\ trials} \tag{5.1}$$

where s is the subject taken into consideration. Additionally, only the two classes, hands and feet, are considered for this accuracy. Therefore, the

number of trials for each subject is fixed at 20, with 10 trials for both hands and 10 for both feet.

- accuracy with rejection. For each trial, we impose a timeout at 21 s in order to finish the trial in at most 21 s to not stress too much the user. Therefore, some trials can reach this time without reaching the hit threshold. Hence the formula is:

$$acc_s = \frac{Number\ of\ hit\ correct}{Number\ of\ sent\ signals} \tag{5.2}$$

  where s is the subject considered and, as before, the trials taken into account are the ones for both hands or both feet.

- time at rest. It is when the user can stay in the INC state and we analyze it when the system asks the user to perform the resting. Additionally, we use two different rests, one that provides feedback to the user and another that does not, respectively *time rest move* and *time rest fixed*.

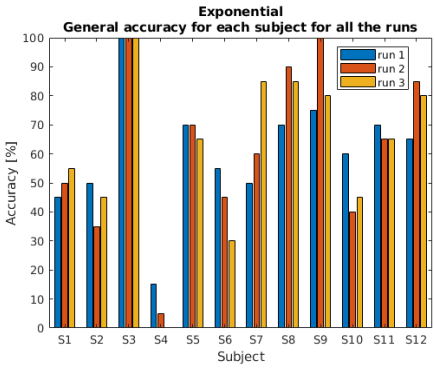
- the results of a questionnaire. Each user at the end of the session must answer a questionnaire to analyze the workload they perceived during the experiment with both control systems. Additionally, we choose the NASA Task Load Index (NASA-TLX) since it has been already validated to assess the user's perceived workload [54, 55].

5.1 Experimental results: first session

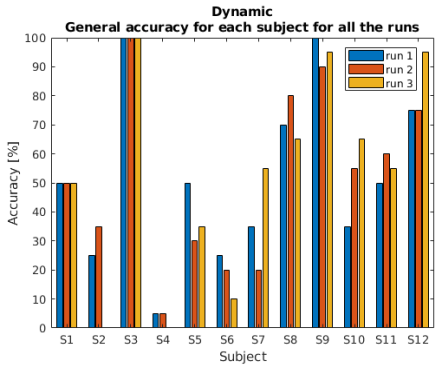
In the first session, the user performs three evaluation runs with the exponential smoothing filter reported in Section 3.1 and three with the dynamic system explained in Section 3.2. All twelve participants took part in this session. Furthermore, the values used for the exponential and the dynamic system are reported in Table 4.1. In this session, as reported in Section 4.1 only a symmetrical behaviour is allowed for both control systems. Moreover, the thresholds used for each subject are fixed at 0.7, while the omega is fixed at 0.2, which is computed as $th - 0.5$ where th is 0.7 for both classes. In addition, for the dynamic control system, we use the relation reported in Figure 3.6 to compute the values for ψ given ω.

Figure 5.2 reports the general accuracy. In detail, Figure 5.2a shows the accuracy for all the subjects with the exponential smoothing filter. As we can see, only five participants reach an accuracy greater than 70% in at least one run. Instead,

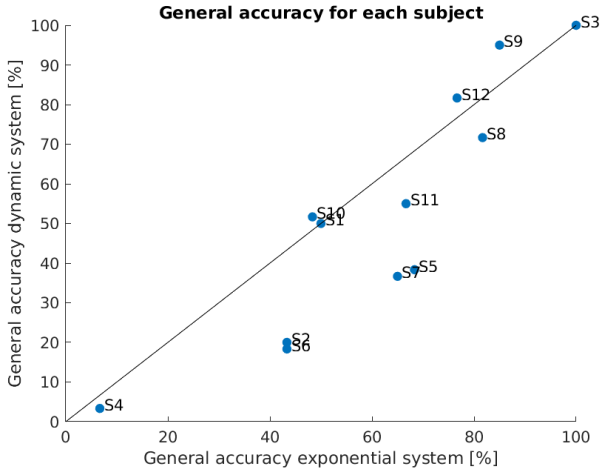
Figure 5.2b shows the accuracy for all the users with the dynamical system. In this case, four subjects reached an accuracy greater than 70%. Additionally, a subject may have an accuracy of 0%, as for S4 with both control systems and S2 with the dynamic. Figure 5.2c confirms that only three subjects obtain, in mean, better results with the dynamical system than with the exponential.



(a) The figure shows the general accuracy of all three runs made by each subject with the exponential control system and the parameters reported in Table 4.1.



(b) The figure shows the general accuracy of all three runs made by each subject with the dynamic control system and the parameters reported in Table 4.1.

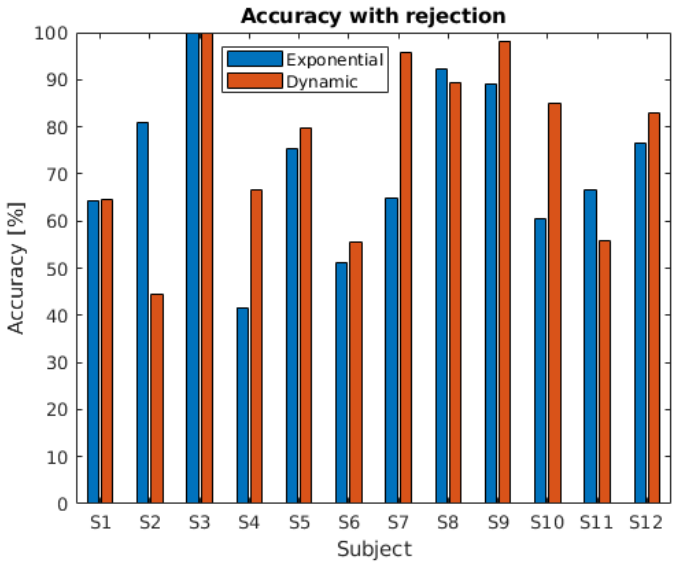


(c) In the figure, each subject is represented by the mean accuracy obtained for both control systems. In addition, a bisector divides the plane into two areas, the upper represents the zone in which the dynamic control system is better than the exponential one, in the lower zone is the contrary.

Figure 5.2: The general accuracy computed with also the trials that reach the timeout in session 1.

Figure 5.3 reports the mean between the three runs of the accuracy with rejection for each subject. As we can see, the dynamic control system provides better results for all the users except for three users: S2, S8, and S11. This is an important factor because it means that the dynamic control system sends fewer wrong signals than the exponential.

To show better the result of the two typologies of accuracy, Figure 5.4 shows that the exponential system provides, in mean, better results if we are talking about general accuracy, instead the dynamic system is better if we look at the



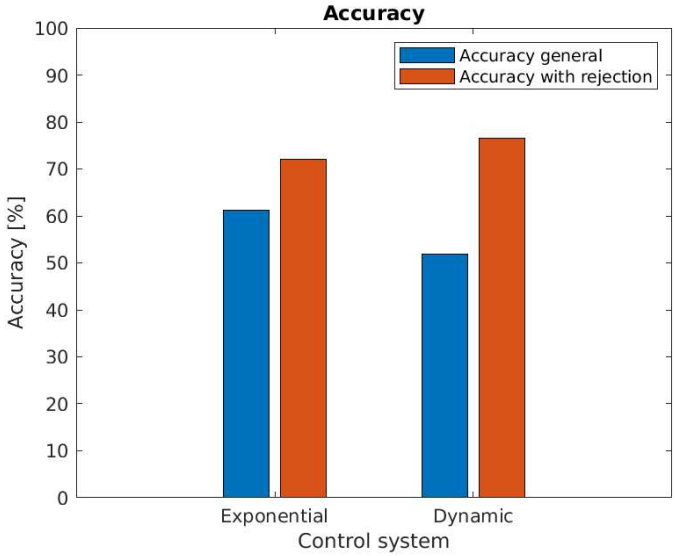
The figure shows the accuracy with rejection for each subject. In addition, each bar is the mean of the three runs of the respective user and control system.

Figure 5.3: accuracy with rejection in session 1.

correctness of the sent signals. Moreover, the exponential system has general accuracy of 61.25%, while removing the trials in which the system reaches the timeout, which corresponds to 18% of the total trials, the accuracy becomes 71.99%. Instead, the dynamic system has a general accuracy of 51.81% and an accuracy with rejection of 76.53% by removing the trials that reach the timeout which is 35.56% of all the trials.

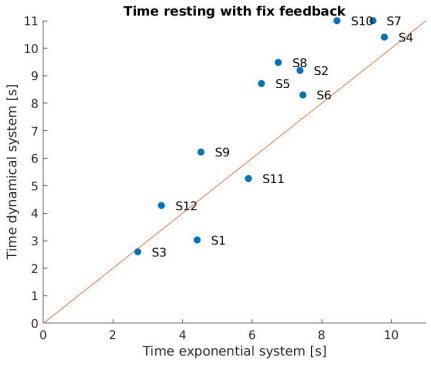
Additionally, we consider the time in which the user can stay in an INC state. Therefore, we analyze the period of the *rest* trials. In other words, we analyze the time in which the user can maintain rest when the system asked to he/she to perform the resting. Furthermore, we can see in Figure 5.5 that for both feedback used (the fixed and the moving), the dynamic control system provides better results. In order to show better the time results we report them in Figure 5.6, in which the exponential filter reaches a time of 6.37s with the fixed feedback and 7.89s with the moving feedback, indeed the dynamic system 7.46s and 10.86s respectively. Therefore, the dynamic control system is better than the exponential one, especially for moving feedback.

The last aspect we analyze is user-perceived workload. Therefore, Figure 5.7 reports the results of the NASA-TLX questionnaire for each subject. In general, seven subjects perceived the dynamic to be heavier than the exponential and six the opposite. One interesting aspect is that users who achieve better performance with dynamic perceive less workload with this control system than

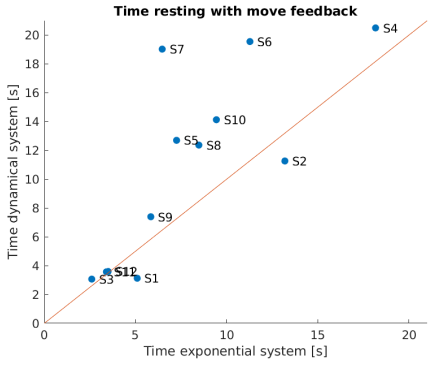


The image shows that the dynamical system provides better results for the accuracy with rejection while the exponential is better if we consider the general accuracy. Therefore, the exponential system obtains 61.25% and 71.99%. Indeed, the dynamical system reaches 51.18% and 76.53%, respectively for general accuracy and accuracy with rejection.

Figure 5.4: The figure reports both the accuracies calculated in session 1.



(a) The figure is reported for each subject the mean time in which a user can stay at rest. Moreover, it considers only fixed feedback. Furthermore, the results show that the dynamic control system is better than the exponential for all subjects except S1, S3 and S11.



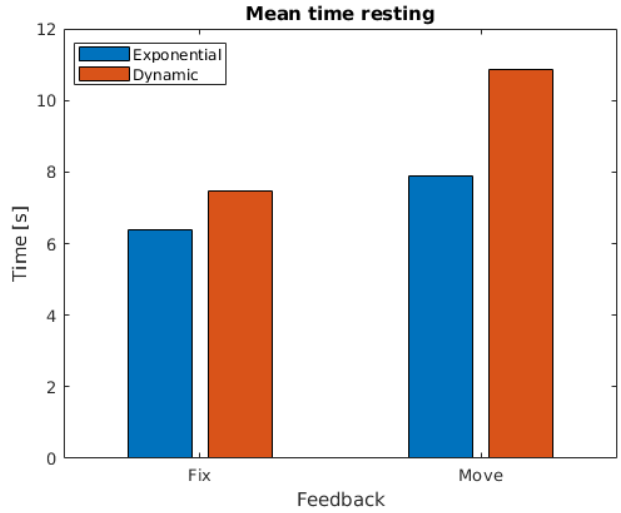
(b) The figure is reported for each subject the mean time in which a user can stay at rest. Moreover, it considers only moving feedback. Furthermore, the results show that the dynamic control system is better than the exponential for all subjects except S1 and S2.

Figure 5.5: Time at rest for each subject in session 1.

with exponential.

5.2 Experimental results: second session

In the second session, the user performs three evaluation runs with the exponential smoothing filter and three with the dynamic system, for both with asymmetrical behaviour, as reported in Section 4.1. Moreover, in this session, we manually select the thresholds used for the exponential control system, on the contrary,



The exponential control system reaches 6.37s with the fixed feedback and 7.89s with the moving feedback. Instead, the dynamic system reaches 7.46s and 10.86s. Therefore, the dynamic control system allows the users to remain for more time in the INC state than the exponential one.

Figure 5.6: The timing results over all subjects for fixed and moving feedback in session 1.

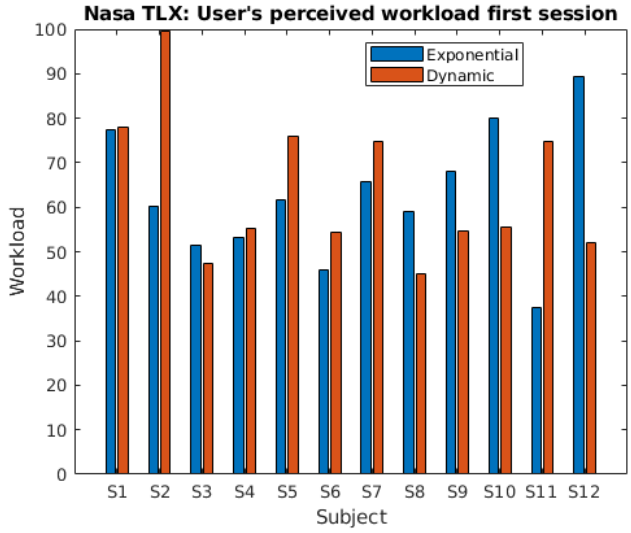


Figure 5.7: NASA-TLX for each subject for the first session

the value of ω_1 and ω_2 are computed according to the relation:

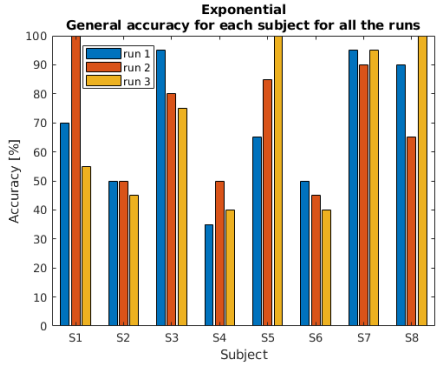
$$\omega = th - 0.5 \tag{5.3}$$

where th is the threshold used in the exponential control system. In addition, for the dynamic control system we use the relation reported in Figure 3.8 to compute the values of ψ_1 and ψ_2 given ω_1 and ω_2.

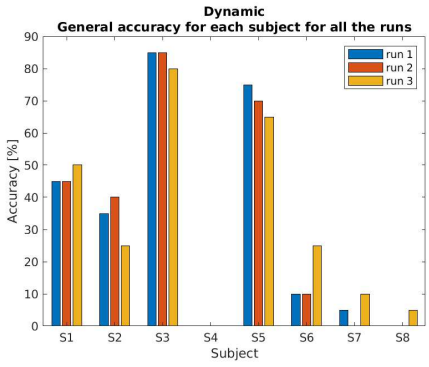
Eight subjects participated in this session and the values used are reported in

Table 4.2.

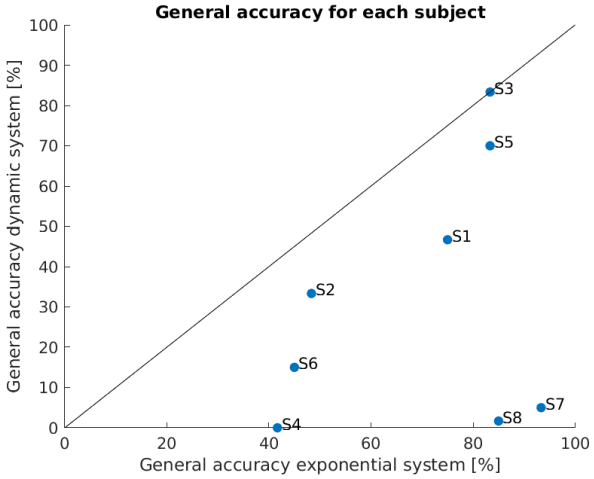
The general accuracy is reported in Figure 5.8. Minutely, Figure 5.8a shows the accuracy for all the subjects with the exponential smoothing filter. In this case, five subjects reach an accuracy greater than 70% in at least one run. On the contrary, Figure 5.8b reports the accuracy obtained with the dynamical system, in this case, only two subjects reach at least 70%. Moreover, subjects S4, S7 and S8 have an accuracy of 0% in at least one trial. Figure 5.8c shows that no subjects obtain, on average, better results with the dynamical system than with the exponential control system. In Section 6 we will discuss these results.



(a) The figure shows the general accuracy of all three runs made by each subject with the exponential control system and the parameters reported in Table 4.2.



(b) The figure shows the general accuracy of all three runs made by each subject with the dynamic control system and the parameters reported in Table 4.2.

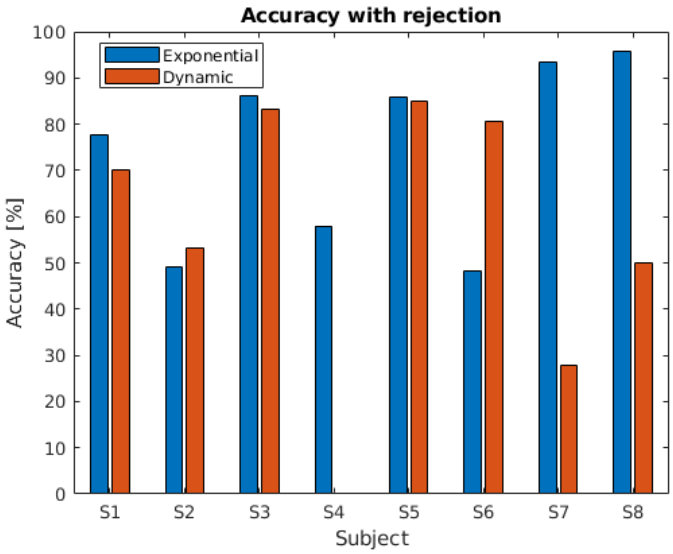


(c) In the figure, each subject is represented by the mean accuracy obtained for both control systems. In addition, a bisector divides the plane into two areas, the upper represents the zone in which the dynamic control system is better than the exponential one, in the lower zone is the contrary.

Figure 5.8: The general accuracy computed with also the trials that reach the timeout in session 2.

Figure 5.9 shows the accuracy with rejection. In this case, only S2 and S6 reach better results with the dynamic control system.

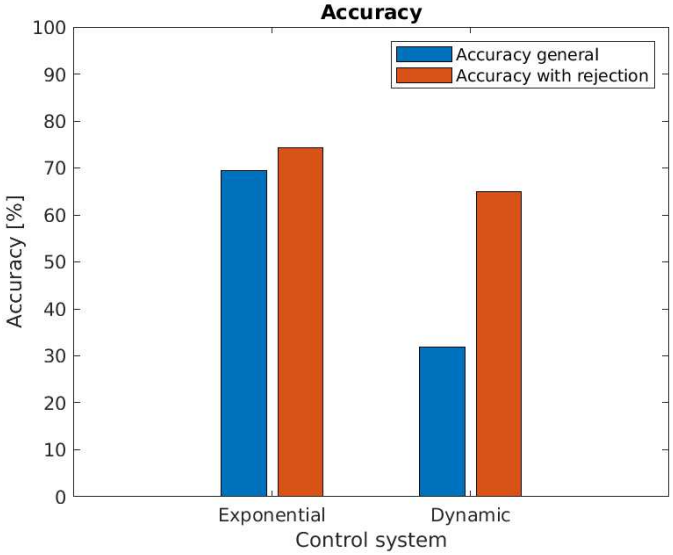
Both accuracies are summarized in Figure 5.10 in which the exponential smoothing filter obtains 69.38% for the general accuracy and 74.27% for the



The figure shows the accuracy with rejection for the dynamical system and the exponential system for all the subjects. Additionally, each bar is the mean of the three runs of the respective user and control system.

Figure 5.9: Accuracy with rejection in session 2.

accuracy with rejection. On the contrary, the dynamic control system reaches 31.88% and 64.99%. Therefore, in this case, the exponential control system has better results than the dynamic for both accuracies.

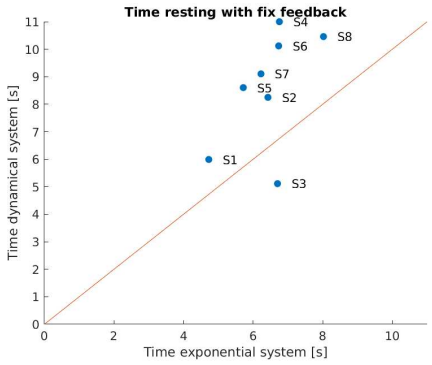


The exponential has obtained a general accuracy of 69.38% while the accuracy with rejection is 74.27%. On the contrary, the dynamic system has 31.88% and 64.99% respectively. Therefore, the exponential control system has better results than the dynamic with both accuracies.

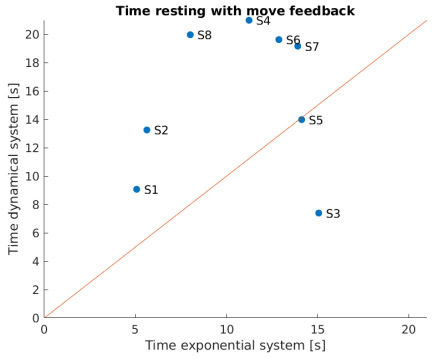
Figure 5.10: The figure reports both the accuracies calculated in session 2.

In addition, we also consider the time in which the user can stay in an INC

state. In other words, we analyze the time in which the user can stay at rest when the system asked to he/she to perform the resting. Furthermore, we can see in Figure 5.11 that for both feedback used (the fixed and the moving), the dynamic control system provides better results. Only S3 have better results with the exponential control system for both the feedback. Figure 5.12 shows better



(a) The figure is reported for each subject the mean time in which a user can stay at rest. Moreover, it considers only fixed feedback. Furthermore, the results show that the dynamic control system is better than the exponential for all subjects except S3.



(b) The figure is reported for each subject the mean time in which a user can stay at rest. Moreover, it considers only moving feedback. Furthermore, the results show that the dynamic control system is better than the exponential for all subjects except S3 and S5.

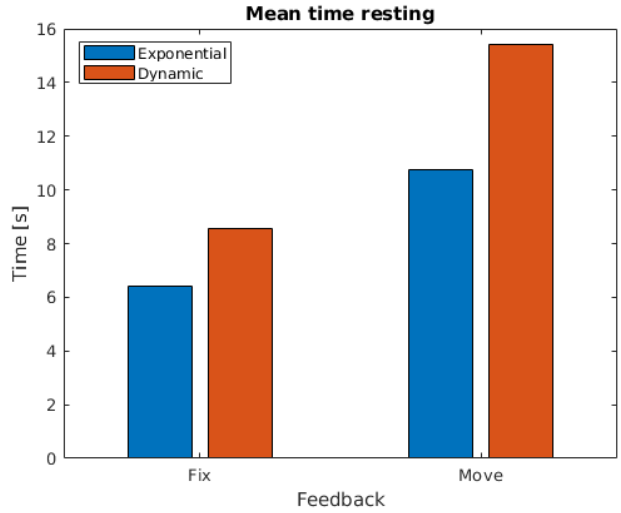
Figure 5.11: Time at rest for each subject in session 2.

the comparison between the exponential and dynamic systems for the fixed and moving feedback. Therefore the users with the exponential control system can stay at rest for 6.42s and 10.74s with fixed and moving feedback respectively. Instead, subjects using the dynamic control system can stay in INC for 8.58s and 15.44s. Therefore, we can see that the dynamic control system is better than the exponential one, especially for the moving feedback.

The last aspect we analyze is user-perceived workload. Figure 5.13 reports the results of the NASA-TLX questionnaire for each subject. In general, all the subjects, except S3, perceive more workload with the dynamic than the exponential. Moreover, S3 reaches better general accuracy with the dynamic control system and he/she feels the same workload with both control systems.

5.3 Experimental results: third session

In the third session, the user performs three evaluation runs with the exponential smoothing filter and three with the dynamic system, for both with asymmetrical behaviour, as reported in Section 4.1. Moreover, in this session, we manually select the thresholds used for the exponential control system and the values of ω_1, ω_2 and ϕ for the dynamic control system. In addition, for the dynamic control



The exponential control system reaches 6.42s with the fixed feedback and 10.74s with the moving feedback. Instead, the dynamic system reaches 8.58s and 15.44s. Therefore, the dynamic control system allows the users to remain more time in the INC state with respect to the exponential.

Figure 5.12: The timing results over all subjects for fixed and moving feedback in session 2.

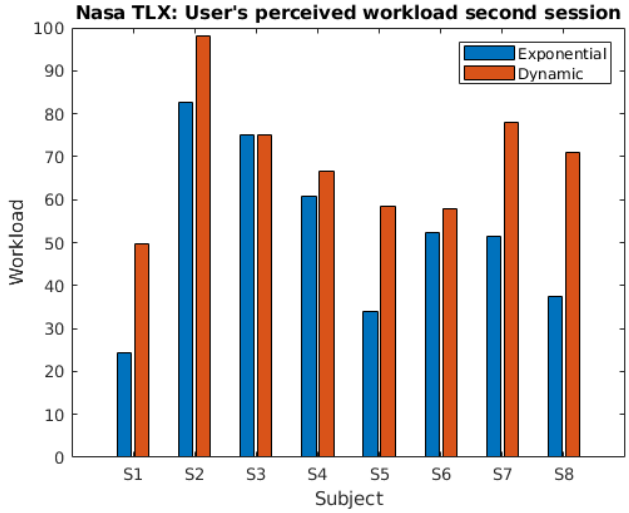
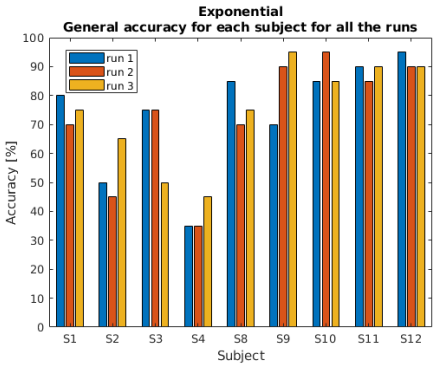


Figure 5.13: NASA-TLX for each subject for the second session

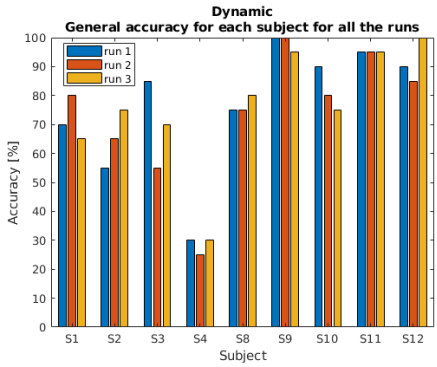
system, we use the relation reported in Figure 3.6 to compute the values of ψ_1 and ψ_2 given ω_1 and ω_2.
Nine subjects participated in this session and the values used are reported in Table 4.3.

Figure 5.14 reports the general accuracy. In detail, Figure 5.14a shows the general accuracy for all the subjects with the exponential smoothing filter. In this case, seven subjects reach an accuracy greater than 70% in at least one run.

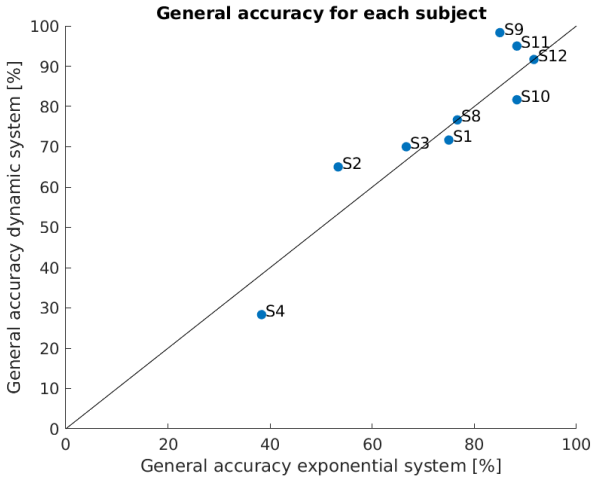
Additionally, the lowest accuracy is reached by S4 and it is 35%. On the contrary, Figure 5.14b reports the accuracy obtained with the dynamical system, in this case, eight subjects reach at least 70%. A big improvement is reached by S2. Figure 5.14c shows that four subjects (S2, S3, S9 and S11) obtain better results with the dynamical system, three (S1, S4, S10) with the exponential system and the last two have the same results with both control systems.



(a) The figure shows the accuracy of all three runs made by each subject with the exponential control system and the parameters reported in Table 4.3.



(b) The figure shows the accuracy of all three runs made by each subject with the dynamic control system and the parameters reported in Table 4.3.

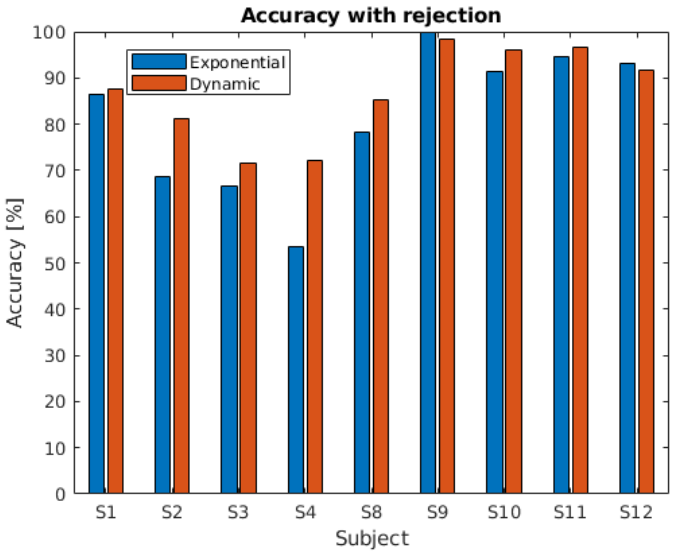


(c) In the figure, each subject is represented by the mean accuracy obtained for both control systems. In addition, a bisector divides the plane into two areas, the upper represents the zone in which the dynamic control system is better than the exponential one, in the lower zone is the contrary.

Figure 5.14: The general accuracy computed with also the trials that reach the timeout in session 3.

Figure 5.15 shows the accuracy over the sent signals. In this case, all subjects reach better results with the dynamic control system except for S9 and S12.

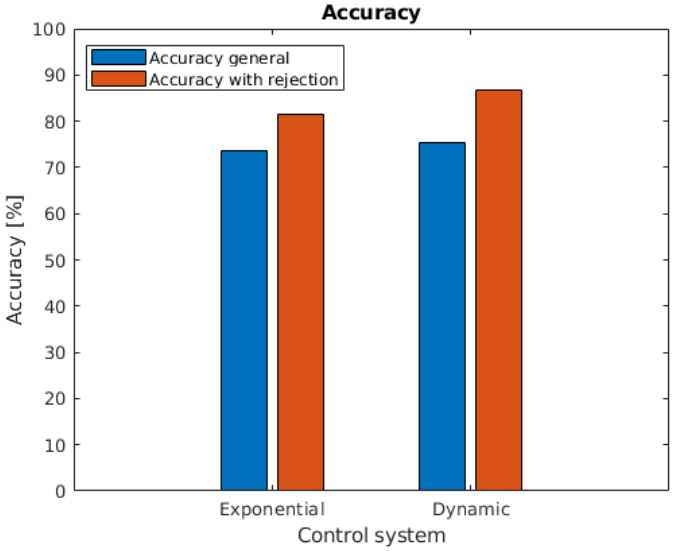
To show better the accuracy results we report them in Figure 5.16. The exponential smoothing filter obtains 73.70% for the general accuracy and 81.46% for the accuracy over all sent signals. On the contrary, the dynamic control system reaches 75.37% and 86.72%. Therefore, in this case, the dynamic control system



The figure shows the accuracy over sent signals for the dynamical system and the exponential system. In addition, each bar is the mean of the three runs of the respective user and control system.

Figure 5.15: Accuracy over the sent signals in session 3.

has better results than the exponential system for both accuracies.

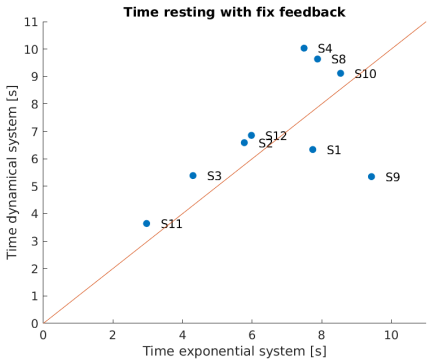


The exponential control system has obtained a general accuracy of 73.70% and an accuracy over all sent signals of 81.46%. On the contrary, the dynamic system reaches 75.37% and 86.72% respectively. Therefore, the dynamic system has better results than the exponential system with both accuracies.

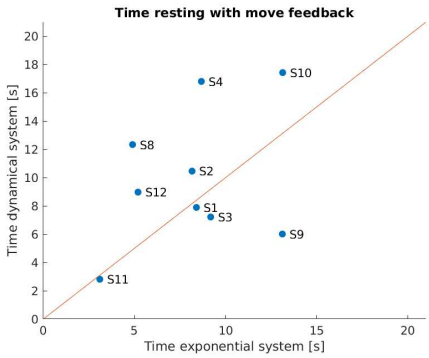
Figure 5.16: The figure reports both the accuracies calculated in session 3.

Additionally, we also consider the time in which the user can stay in an INC state. In other words, we analyze the time in which the user can stay at rest when the system asked to he/she to perform the resting. Furthermore, we can see in Figure 5.17 that for both feedback used (the fixed and the moving) that

the dynamic control system provides better results. In detail, Figure 5.17a, which represents the time for the fixed feedback, shows that the dynamic control system is better than the exponential for all the subjects except S1 and S9. In addition, the moving feedback, reported in Figure 5.17b, shows that five subjects perform better with the dynamic control system and four with the exponential one. Figure



(a) The figure reports for each subject the mean time in which a user can stay at rest. Moreover, it considers only fixed feedback. Furthermore, the results show that the dynamic control system is better than the exponential for all subjects except S1 and S9.

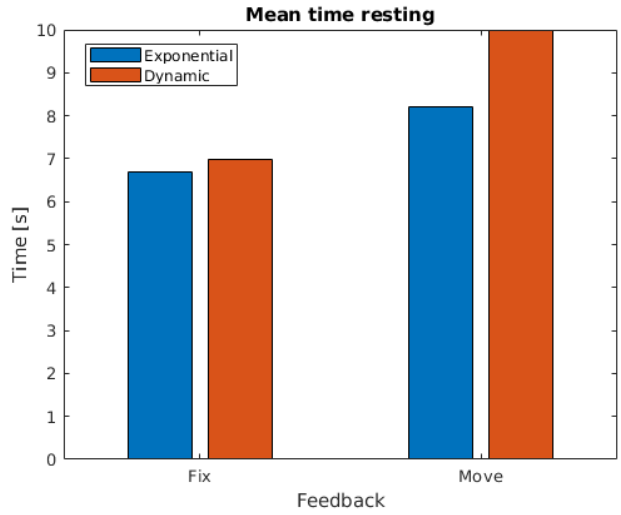


(b) The figure reports for each subject the mean time in which a user can stay at rest. Moreover, it considers only moving feedback. Furthermore, the results show that S1, S3, S9 and S11 can stay more time at rest with the exponential control system, while S2, S4, S8, S10 and S12 with the dynamic control system.

Figure 5.17: Time at rest for each subject in session 3.

5.18 shows better the comparison between the exponential and dynamic systems for the fixed and moving feedback. Therefore the users with the exponential control system can stay at rest for 6.68s and 8.21s with fixed and moving feedback respectively. Instead, subjects using the dynamic control system can stay in INC for 7.00s and 10.00s. Therefore, we can see that the dynamic control system is better than the exponential one.

The last aspect we analyze is user-perceived workload. Figure 5.19 reports the results of the NASA-TLX questionnaire for each subject. In general, all the subjects, except S4, perceive less workload with the dynamic than the exponential. This confirms our hypothesis that an asymmetric dynamic control system provides less workload to the user.



The exponential control system reaches 6.68s with the fixed feedback and 8.21s with the moving feedback. On the contrary, the dynamic system reaches 7.00s and 10.00s. Therefore, the dynamic control system provides better results than the exponential one.

Figure 5.18: The timing results over all subjects for fixed and moving feedback in session 3.

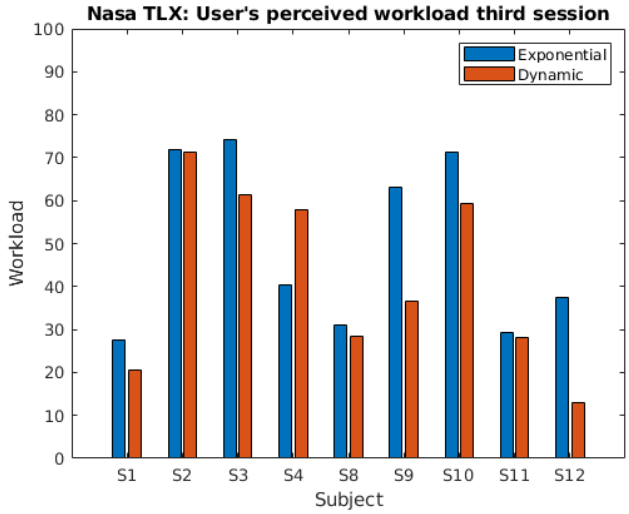


Figure 5.19: NASA-TLX for each subject for the third session

Chapter 6

Discussion

In this chapter, we will analyze the results reported in Chapter 5. As highlighted in Section 1.5.1 these experiments are performed to validate our initial hypotheses: there is a relation between ω and ψ, and the validation of the asymmetrical dynamic system. In addition, we need to perform these three experiment sessions because the BMI system is strictly user depend, so we need to validate our hypotheses through real experiments and not with simulated ones.

In the first session, we use both control systems symmetrically. Therefore, for all the subjects, the same thresholds are used for both hands and both feet and are at 0.7. Also, the dynamic control system works symmetrically with the same ω that is fixed at $\omega = 0.2$ because we follow the relation $\omega = threshold - 0.5$. In this way, the valley's peaks shown in Figure 3.6b are at 0.7 and 0.3 which are the thresholds used for the exponential control system, using only one raw probability since we have only two classes.
The results reported in Section 5.1 show that the exponential is more accurate in the general case since it reaches 61.25% while the dynamic obtains 51.18%. Instead, the dynamic control system has better accuracy over sent signals 76.53% over the 71.99% obtained by the exponential one. This shows that the dynamic control system allows users to send fewer wrong control signals.
Looking at the time in INC, the dynamic control system is higher for fixed and moving feedback than the exponential one.
Therefore, with this experiment, we confirm what was reported also in [15] because the accuracy is pretty similar for both cases and the time in INC is better for the dynamic control system.

In the second session, we use both control systems asymmetrically. In this case, we manually select the thresholds used for the exponential control system, while keeping the same reasoning used in the previous session, we compute ω_1 =threshold both hands $-$ 0.5 and ω_2 = threshold both feet -0.5; for all subject the parameters are reported in Table 4.2. In addition, we use the relation in Figure 3.8 to calculate the other parameters for the dynamic control system.
The results reported in Section 5.2 show that the exponential control system is more accurate than the dynamic for both accuracies calculated. In addition, looking at the time the dynamic is better than the exponential by more or less 2 s for the fixed feedback and 5 s for the moving feedback.
These results do not provide that the dynamic control system is better than the exponential one especially, for accuracy. Therefore, we hypothesise:

- The asymmetric relation is wrong. This is because with a too-high ψ the subject cannot overtake the potential valley (Figure 3.3b), therefore for the user is too difficult to send a command. This is also visible in the huge difference in the timing of rest with both feedback.
- The relation that we hypothesize exists between ω_1, ω_2 and the respective threshold of the exponential control system is wrong.
- We manually select the best parameter for the exponential control system, but not for the dynamic. Therefore, we favour exponential control over dynamic control because we found pseudo-optimum values for it and not for the other.

Therefore we proposed some corrections for the third session:

- Use the symmetrical relation also for the asymmetrical dynamic system. Since it allows having smaller ψs values.
- Manually select the value of ω_1, ω_2 and ϕ for the dynamic control system as for the thresholds of the exponential control system, based on the user's perception of controlling the BMI.

In order to validate our hypotheses we perform a third session.

In the third session, we use both control systems asymmetrically, we select manually all the required values for the parameters, which are reported in Table 4.3.

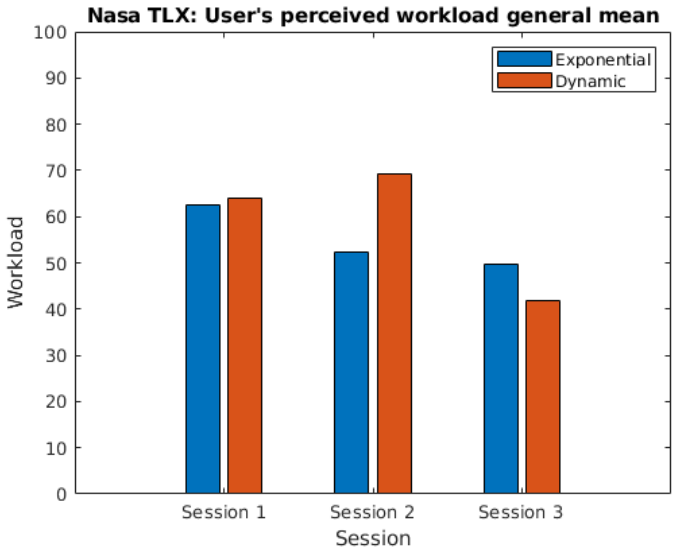


Figure 6.1: NASA-TLX in general for each session

The results reported in Section 5.3 confirm our hypothesis. Also, the hypotheses formulated to understand the second session are correct since following them we have that the dynamic control system allows users to stay more in INC state for both feedback, as reported in Figure 5.18 and to be more accurate as shown in Figure 5.16.
Therefore, this experiment proves that the relation exposed in Figure 3.6 can be used also for the asymmetric dynamic control system and that this control system is better than the exponential one.

In general, looking at the workload perceived by the users in each session, as reported in Figure 6.1, it can be seen that the assumptions made are correct. Moreover, in the first session, the workload perceived with the dynamic system is very close to the one perceived with the exponential system. On the contrary, in the second session, there is a huge difference between the two systems due to the wrong relations used. Instead, with the corrections made in the third session the users perceived less workload with the dynamic system than with the exponential one.

Chapter 7

Conclusion

The main concepts of this thesis are regarding a relation between parameters used in the F_{free} formulation and the asymmetrical implementation of the dynamic system for a BMI system based on 2-class motor imagery.
The results of the first and third sessions confirm that a relation between ω and ψ can be the one in Figure 3.6 for the symmetrical and asymmetrical case. On the contrary, the second session demonstrates that the relation reported in Figure 3.8 provides too high ψs and the user cannot overtake the potential valley created by the free force (Figure 3.3b). Therefore, this correlation was discarded.
The main contribution of this thesis to the state-of-art is regarding the implementation and validation, through some experiments, of an asymmetrical dynamic system with a relation that allows the reduction of the number of required parameters for a brain-machine interface system based on 2-class motor imagery.

However, some observations about this work can be done. First of all, the validation is limited since we have a small number of subjects, hence future work can increase the number of people involved. Secondly, the dynamical control system provides better results, especially when the user is in control of an external device, as reported in [15]. Therefore, we can further validate this asymmetric control system for driving a telepresence robot or a powered wheelchair.

References

[1] Chaudhary U, Birbaumer N, and Ramos-Murguialday A. Brain–computer interfaces for communication and rehabilitation. *Nature Reviews Neurology*, 12(9):513–525, 2016.

[2] McFarland D J and Wolpaw J R. Brain-computer interfaces for communication and control. *Communications of the ACM*, 54(5):60–66, 2011.

[3] Tonin L and Millán JdR. Noninvasive brain–machine interfaces for robotic devices. *Annual Review of Control, Robotics, and Autonomous Systems*, 4(1):191–214, 2021.

[4] Birbaumer N, Ghanayim N, Hinterberger T, Iversen I, Kotchoubey B, Kübler A, Perelmouter J, Taub E, and Flor H. A spelling device for the paralysed. *Nature*, 398(6725):297–298, 1999.

[5] Donchin E, Spencer K, and Wijesinghe R. The mental prosthesis: Assessing the speed of a p300-based brain-computer interface. *IEEE transactions on rehabilitation engineering*, 8:174–9, 2000.

[6] Ferguson K A, Polando G, Kobetic R, Triolo R J, and Marsolais E B. Walking with a hybrid orthosis system. *Spinal Cord*, 37(11):800–804, 1999.

[7] He B, Han Y, Jianjun M, and Shangkai G. Brain–computer interfaces. *Neural Engineering*, pages 131–183, 2020.

[8] Millán JdR, Rupp R, Mueller-Putz G, Murray-Smith R, Giugliemma C, Tangermann M, Vidaurre C, Cincotti F, Kubler A, Leeb R, Neuper C, Mueller K, and Mattia D. Combining brain–computer interfaces and assistive technologies: State-of-the-art and challenges. *Frontiers in Neuroscience*, 4, 2010.

[9] Waldert S, Pistohl T, Braun C, Ball T, Aertsen A, and Mehring C. A review on directional information in neural signals for brain-machine interfaces. *Journal of Physiology-Paris*, 103(3):244–254, 2009.

[10] Schalk G, Kubánek J, Miller K J, Anderson N R, Leuthardt E C, Ojemann J G, Limbrick D, Moran D, Gerhardt L A, and Wolpaw J R. Decoding two-dimensional movement trajectories using electrocorticographic signals in humans. *Journal of Neural Engineering*, 4(3):264–275, 2007.

[11] Weiskopf N, Mathiak K, Bock S W, Scharnowski F, Veit R, Grodd W, Goebel R, and Birbaumer N. Principles of a brain-computer interface (bci) based on real-time functional magnetic resonance imaging (fmri). *IEEE Transactions on Biomedical Engineering*, 51(6):966–970, 2004.

[12] Coyle S M, Ward T E, and Markham C M. Brain–computer interface using a simplified functional near-infrared spectroscopy system. *Journal of Neural Engineering*, 4(3):219, 2007.

[13] Zhijun L, Suna Z, Jiding D, Chun-Yi S, Chenguang Y, and Xingang Z. Human cooperative wheelchair with brain–machine interaction based on shared control strategy. *IEEE/ASME Transactions on Mechatronics*, 22(1):185–195, 2017.

[14] Iturrate I, Antelis J M, Kubler A, and Minguez J. A noninvasive brain-actuated wheelchair based on a p300 neurophysiological protocol and automated navigation. *IEEE Transactions on Robotics*, 25(3):614–627, 2009.

[15] Tonin L, Bauer F C, and José JdR. The role of the control framework for continuous teleoperation of a brain–machine interface-driven mobile robot. *IEEE Transactions on Robotics*, 36(1):78–91, 2020.

[16] Beraldo G, Tortora S, Menegatti E, and Tonin L. Ros-neuro: implementation of a closed-loop bmi based on motor imagery. *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2031–2037, 2020.

[17] Beraldo G, Tonin L, Millán JdR, and Menegatti E. Shared intelligence for robot teleoperation via bmi. *IEEE Transactions on Human-Machine Systems*, pages 1–10, 2022.

[18] Yiliang L, Wenbin S, Zhijun L, Guangming S, Xiaoli C, Yu K, and Weiwei S. Motor-imagery-based teleoperation of a dual-arm robot performing manipulation tasks. *IEEE Transactions on Cognitive and Developmental Systems*, 11(3):414–424, 2019.

[19] Doud A J, Lucas J P, and He B. Continuous 3d control of a virtual helicopter using a motor imagery based bci. *2011 5th International IEEE/EMBS Conference on Neural Engineering*, pages 364–367, 2011.

[20] LaFleur K, Cassady K, Doud A, Shades K, Rogin E, and He B. Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface. *Journal of Neural Engineering*, 10(4):046003, 2013.

[21] Satti A, Coyle D, and Prasad G. Continuous eeg classification for a self-paced bci. *2009 4th International IEEE/EMBS Conference on Neural Engineering*, pages 315–318, 2009.

[22] Coyle D, Garcia J, Satti A R, and McGinnity T M. Eeg-based continuous control of a game using a 3 channel motor imagery bci: Bci game. *2011 IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB)*, pages 1–7, 2011.

[23] Yuanqing L, Jiahui P, Fei W, and Zhuliang Y. A hybrid bci system combining p300 and ssvep and its application to wheelchair control. *IEEE Transactions on Biomedical Engineering*, 60(11):3156–3166, 2013.

[24] Beraldo G, Castaman N, Bortoletto R, Pagello E, Millán JdR, Tonin L, and Menegatti E. Ros-health: An open-source framework for neurorobotics. *2018 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR)*, pages 174–179, 2018.

[25] Tonin L, Beraldo G, Tortora S, Tagliapietra L, Millán JdR, and Menegatti E. Ros-neuro: A common middleware for bmi and robotics. the acquisition and recorder packages. *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 2767–2772, 2019.

[26] Perdikis S, Tonin L, Saeedi S, Schneider C, and Millán JdR. The cybathlon bci race: Successful longitudinal mutual learning with two tetraplegic users. *PLOS Biology*, 16:1–28, 05 2018.

[27] Amiri S, Rabbi A, Azinfar L, and Fazel-Rezai R. A review of p300, ssvep, and hybrid p300/ssvep brain- computer interface systems. *Brain-Computer Interface Systems*, 2013.

[28] Stawicki P, Gembler F, and Volosyak I. Driving a semiautonomous mobile robotic car controlled by an ssvep-based bci. *Computational Intelligence and Neuroscience*, 2016:4909685, 2016.

[29] Mueller S, Cardoso W, Freire T, and Sarcinelli-Filho M. Brain-computer interface based on visual evoked potentials to command autonomous robotic wheelchair. *J. Med. Biol. Eng*, 30, 2010.

[30] McCarthy G and Donchin E. A metric for thought: A comparison of p300 latency and reaction time. *Science*, 211(4477):77–80, 1981.

[31] Farwell L A and Donchin E. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6):510–523, 1988.

[32] Lopes A C, Pires G, and Nunes U. Assisted navigation for a brain-actuated intelligent wheelchair. *Robotics and Autonomous Systems*, 61(3):245–258, 2013.

[33] Shenghong H, Rui Z, Qihong W, Yang C, Tingyan Y, Zhenghui F, Yuandong Z, Ming S, and Yuanqing L. A p300-based threshold-free brain switch and its application in wheelchair control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(6):715–725, 2017.

[34] Brice R, Cuntai G, Haihong Z, Chuanchu W, Cheeleong T, Marcelo A H, and Etienne B. A brain controlled wheelchair to navigate in familiar environments. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(6):590–598, 2010.

[35] Pfurtscheller G. Functional brain imaging based on erd/ers. *Vision Research*, 41(10):1257–1260, 2001.

[36] Pfurtscheller G and Neuper C. Motor imagery and direct brain-computer communication. *Proceedings of the IEEE*, 89(7):1123–1134, 2001.

[37] Vanacker G, Millán JdR, Lew E, Ferrez P W, Moles FG, Philips J, Van Brussel H, and Nuttin M. Context-based filtering for assisted brain-actuated wheelchair driving. *Computational Intelligence and Neuroscience*, 2007:025130, 2007.

[38] Yi-Qian H, Tian-Hao G, Jie L, Jia-Chao T, Yu-Long B, and Rong-Rong L. Motor imagery-based brain-computer interface combined with multimodal feedback to promote upper limb motor function after stroke: A preliminary study. *Evidence-Based Complementary and Alternative Medicine*, 2021:1116126, 2021.

[39] Keun-Tae K, Carlson T, and Seong-Whan L. Design of a robotic wheelchair with a motor imagery based brain-computer interface. *2013 International Winter Workshop on Brain-Computer Interface (BCI)*, pages 46–48, 2013.

[40] Bhattacharyya S, Shimoda S, and Hayashibe M. A synergetic brain-machine interfacing paradigm for multi-dof robot control. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(7):957–968, 2016.

[41] Lécuyer A, Lotte F, Reilly R B, Leeb R, Hirose M, and Slater M. Brain-computer interfaces, virtual reality, and videogames. *Computer*, 41(10):66–72, 2008.

[42] Kuhner D, Fiederer L D J, Aldinger J, Burget F, Völker M, Schirrmeister R T, Do C, Boedecker J, Nebel B, Ball T, and Burgard W. A service assistant combining autonomous robotics, flexible goal formulation, and deep-learning-based brain–computer interfacing. *Robotics and Autonomous Systems*, 116:98–113, 2019.

[43] Rui Z, Yuanqing L, Yongyong Y, Hao Z, Shaoyu W, Tianyou Y, and Zhenghui G. Control of a wheelchair in an indoor environment based on a brain–computer interface and automated navigation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(1):128–139, 2016.

[44] Yang X, Ding C, Shu X, Gui K, Bezsudnova Y, Sheng X, and Zhang D. Shared control of a robotic arm using non-invasive brain–computer interface and computer vision guidance. *Robotics and Autonomous Systems*, 115:121–129, 2019.

[45] Wenchang Z, Fuchun S, Chunfang L, Weihua S, Chuanqi T, and Shaobo L. A hybrid eeg-based bci for robot grasp controlling. *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3278–3283, 2017.

[46] Meng J, Zhang S, Bekyo A, Olsoe J, Baxter B, and He B. Noninvasive electroencephalogram based control of a robotic arm for reach and grasp tasks. *Scientific Reports*, 6(1):38565, 2016.

[47] Mason S G, Bashashati A, Fatourechi M, Navarro K F, and Birch G E. A comprehensive survey of brain interface technology designs. *Annals of Biomedical Engineering*, 35(2):137–169, 2007.

[48] Leeb R, Perdikis S, Tonin L, Biasiucci A, Tavella M, Creatura M, Molina A, Al-Khodairy A, Carlson T, and Millán JdR. Transferring brain–computer interfaces beyond the laboratory: Successful application control for motor-disabled users. *Artificial Intelligence in Medicine*, 59(2):121–132, 2013.

[49] Philips J, Millan JdR, Vanacker G, Lew E, Galan F, Ferrez P W, Van Brussel H, and Nuttin M. Adaptive shared control of a brain-actuated simulated wheelchair. *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pages 408–414, 2007.

[50] Leeb R, Perdikis S, Tonin L, Biasiucci A, Tavella M, Creatura M, Molina A, Al-Khodairy A, Carlson T, and Millán JdR. Transferring brain–computer interfaces beyond the laboratory: Successful application control for motor-disabled users. *Artificial Intelligence in Medicine*, 59(2):121–132, 2013.

[51] Leeb R, Tonin L, Rohm M, Desideri L, Carlson T, and Millán JdR. Towards independence: A bci telepresence robot for people with severe motor disabilities. *Proceedings of the IEEE*, 103(6):969–982, 2015.

[52] Tonin L, Beraldo G, Tortora S, and Menegatti E. Ros-neuro: An open-source platform for neurorobotics. *Frontiers in Neurorobotics*, 16, 2022.

[53] Leeb R, Tonin L, Rohm M, Desideri L, Carlson T, and Millán JdR. Towards independence: A bci telepresence robot for people with severe motor disabilities. *Proceedings of the IEEE*, 103(6):969–982, 2015.

[54] Hart S G and Staveland L E. Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Human Mental Workload*, 52:139–183, 1988.

[55] Hart S G. Nasa-task load index (nasa-tlx); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9):904–908, 2006.