2017

Error-related potentials for adaptive decoding and volitional control

BOSTON UNIVERSITY

SCHOOL OF MEDICINE

Dissertation

ERROR-RELATED POTENTIALS FOR ADAPTIVE DECODING AND VOLITIONAL CONTROL

by

ANDRÉS FELIPE SALAZAR GÓMEZ

B.Sc., Escuela de Ingeniería de Antioquia-Universidad CES, 2006

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

2017

Approved by

First Reader ______________________________

Frank H. Guenther, Ph.D.
Professor Department of Speech, Language and Hearing Sciences

Second Reader ______________________________

Daniel H. Bullock, Ph.D.
Professor Department of Psychological & Brain Sciences

Third Reader ______________________________

Deniz Erdogmus, Ph.D.
Associate Professor, Electrical and Computer Engineering
Northeastern University, College of Engineering

"Que otros se jacten de las páginas que han escrito;

a mi me enorgullecen las que he leído"

Jorge Luis Borges

"Hay pocas cosas tan ensordecedoras como el silencio"

Mario Benedetti

DEDICATION

I dedicate this work to my parents and sister:

For being the beacon that has always lit my way.

ACKNOWLEDGMENTS

The work described in this dissertation could have not been done without the collaboration of the members of the Neural Prosthesis Lab, the Speech Lab, the Distributed Robotics Lab and the Miller Lab. Our daily interactions proved to be an infinite source of ideas and inspiration, and fueled the spark that lit the way to completing my degree. I will do my best to recognize their participation.

I had the opportunity of learning from several outstanding mentors. My advisor Frank Guenther always looked for the best way to channel my interests and helped me foster exciting collaborations whilst keeping me focused and on track. With Jonathan Brumberg I started my first project in brain-computer interfaces (BCI) and under his constant support and feedback I opened my eyes to the possibilities this field has to offer. The first experiences in BCI shaped the rest of my Ph.D. Everything I know about electrophysiology and invasive recordings is thanks to Scott Brincat's patient mentorship. Moreover, his insights into data analysis, code structure and paradigm design will accompany me forever. Finally, Daniela Rus opened to me the doors of her lab and allowed me to combine robotics and neuroscience. I will always be grateful for her continuous encouragement and passion in our project.

I also want to thank Nan Jia and Misha Panko with whom I spent months and years collecting data, sharing ideas, discussing our project, and results. From all the time we spent together I grew not only professionally but personally. Many thanks go too to Ayoub Daliri, Saul Frankford and Austin Soplata who were always in the lab to share a moment of conversation in between analysis or to discuss results and possible solutions to my research problems. From our exchanges I learned that a Ph.D. is about sharing ideas and interests with others; joining forces is what truly allows the magic of big things to happen.

I also want to thank:

- The other members of my Dissertation Advisory Committee: Jason Ritt, Dan Bullock and Deniz Erdogmus. Your feedback and knowledgeable insights made this journey possible.
- The members of the DSL lab, especially Joseph DelPreto and Bianca Homberg, who made possible working with Baxter.
- Barbara Holland “Bobbie”, who was always available to rescue me upon administrative and grammatical mayhem.
- Other members of the Speech Lab: Emily Stephen, Dante Smith, Spencer Torene, Byron Galbraith, Elisa Golfinopoulos, Jenn Segawa and Jason Tourville.
- Shelley Russek and Sandi Grasso
- GPN, CNS, CompNet, CNSO
- My funding: CELEST NSF SMA-0835976

Finally, I have special gratitude towards Stephanie: for being an endless source of inspiration and support; for always pushing me beyond my limits and for unconditionally believing in me.

ERROR-RELATED POTENTIALS

FOR ADAPTIVE DECODING AND VOLITIONAL CONTROL

ANDRÉS FELIPE SALAZAR GÓMEZ

Boston University School of Medicine, 2017

Major Professor: Frank H. Guenther Ph.D., Professor, Department of Speech, Language and Hearing Sciences

ABSTRACT

Locked-in syndrome (LIS) is a condition characterized by total or near-total paralysis with preserved cognitive and somatosensory function. For the locked-in, brain-machine interfaces (BMI) provide a level of restored communication and interaction with the world, though this technology has not reached its fullest potential. Several streams of research explore improving BMI performance but very little attention has been given to the paradigms implemented and the resulting constraints imposed on the users. Learning new mental tasks, constant use of external stimuli, and high attentional and cognitive processing loads are common demands imposed by BMI. These paradigm constraints negatively affect BMI performance by locked-in patients. In an effort to develop simpler and more reliable BMI for those suffering from LIS, this dissertation explores using error-related potentials, the neural correlates of error awareness, as an access pathway for adaptive decoding and direct volitional control.

In the first part of this thesis we characterize error-related local field potentials (eLFP) and implement a real-time decoder error detection (DED) system using eLFP while non-human primates controlled a saccade BMI. Our results show specific traits in the eLFP that bridge current knowledge of non-BMI evoked error-related potentials with

error-potentials evoked during BMI control. Moreover, we successfully perform real-time DED via, to our knowledge, the first real-time LFP-based DED system integrated into an invasive BMI, demonstrating that error-based adaptive decoding can become a standard feature in BMI design.

In the second part of this thesis, we focus on employing electroencephalography error-related potentials (ErrP) for direct volitional control. These signals were employed as an indicator of the user's intentions under a closed-loop binary-choice robot reaching task. Although this approach is technically challenging, our results demonstrate that ErrP can be used for direct control via binary selection and, given the appropriate levels of task engagement and agency, single-trial closed-loop ErrP decoding is possible.

Taken together, this work contributes to a deeper understanding of error-related potentials evoked during BMI control and opens new avenues of research for employing ErrP as a direct control signal for BMI. For the locked-in community, these advancements could foster the development of real-time intuitive brain-machine control.

PREFACE

Imagine your mind trapped in a crystal castle, in a prison filled with sounds, images and feelings. Imagine your body far, far away, yet next to you... in the darkness, chained to a bed and waiting for the pale rider. No words come out of this prison; your thoughts stuck in your mind, bouncing back and forth, without an exit...

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AAC……………………………………......Augmentative and alternative communication

ACC……………………………………...…………...…….Anterior cingulate cortex

ACM……………………………………...……………Average covariance matrix

AD…………………………………………...…………………Adaptive decoding

ALDA……………………………………………Adaptive linear discriminant analysis

ANOVA……………………….…………………...……….……Analysis of variance

ANS……………………….………………...…..………..Autonomic nervous system

AUC…………………………………………...………….….Area under the curve

BCI……………………….………………………………Brain-computer interface

BMI……………………………………………………….……Brain-machine interface

BOLD……………………...………………………………Blood-oxygen-level dependent

DED……………………...……………………………….….Decoder error detection

CAR…………………...………………………………………Common average reference

CI…………………………………………………………………Confidence interval

CSP………………...……………………………………………Common spatial patterns

CV……………………………………………………………...…….Cross-validation

ECoG………………………………………………….................Electrocorticography

EEG…………….……………………………………………….…..Electroencephalography

eLFP…………………………………………………Error-related local field potentials

ErrDiff………...…………………………………………………......Error difference

ErrP ……………………………………...Electroencephalography error-related potentials

FA………...Focused attention

FEF……………………………………………………………..Frontal eye field

fMRI………………..Functional magnetic resonance imaging

FP………………………………………………….………......False positive rate

HRI……...………………………………………………….Human–robot interaction

iErrP…...……………….……Interaction electroencephalography error-related potentials

LFP………………………………………………………...Local field potentials

LIS…………………………………...…………………………Locked-in syndrome

MEG……………………………...……………….…..Magnetoencephalography

MDM….…………………………......…………….Riemann minimum distance to mean

MSE……………………………...…………………...……….Minimum square error

Ne……………………………...……………..…Error-related negativity/error negativity

NIRS…………………………...………………..……..........Near infrared spectroscopy

NHP…………………………...……………………………………Non-human primate

oErrP……………………......Observation electroencephalography error-related potentials

OM……………………………………………………………………..Open monitoring

PCA………………………………………………………Principal component analysis

Pe…………………………...…………………………………………Error positivity

PFC……………………...………………………………..Dorsolateral prefrontal cortex

pMFC…………………...………………………………….Posterior medial frontal cortex

pre-SMA……………...…………………………………....Pre-supplementary motor area

PTB……………………………………………………………………PsychToolBox

REFSF…………………………………………Fisher event-related potential spatial filter

ROC…...………………………………………………Receiver operating characteristic

RSVP………………………………………..........…Rapid serial visual presentation

SCM……………………………………………..………..Sample covariance matrix

S.D………………………………………...……………………Standard deviation

SEF…………………………………………………………..Supplementary eye field

SNR……………………………………...……….…………..….Signal-to-noise ratio

SSVEP………………………………...…...…...Steady state visually-evoked potentials

SVM……………………………...…...………………...Support vector machine

TP……………………………………………………………..……….True positive rate

TN…………………………………...……………………………..True negative rate

WTCSP…………………………...…...Weighted Tikhonov common spatial patterns

CHAPTER I: INTRODUCTION

Brain-machine interfaces (BMI) have the potential to restore the quality of life of people with disabilities, in particular those in the locked-in state, a condition characterized by quadriplegia and anarthria but with largely preserved cognitive functions (Bauer et al., 1979; Beaudoin & De Serres, 2010). In spite of their progress, current invasive and non-invasive technologies do not meet the locked-in needs due to technical and physiological factors (Birbaumer, 2006a). Improving BMI performance has become an active area of research focused mainly on increasing the signal quality (McFarland et al., 1997; Parra et al., 2008), exploring new paradigms and channels of communication (Kushki et al., 2012; Millán et al., 2010), and implementing adaptive decoding (Dangi et al., 2013). Nevertheless, the locked-in still face big challenges for controlling BMI (Chaudhary et al., 2016).

In an effort to develop simpler and more reliable BMI paradigms for those suffering from locked-in syndrome, this dissertation explores error-related potentials, the neural correlates of error awareness, as an access pathway for adaptive decoding and volitional control. We demonstrate that decoder errors i) can be classified in real-time using intra-cortical error-related potentials in an invasive BMI, and ii) we propose and implement a BMI paradigm that relies only on non-invasive error-related potentials for direct binary-choice decoding. This chapter presents relevant literature to the project and summarizes the contents of the dissertation.

1. Locked-in syndrome (LIS)

Locked-in syndrome (LIS) is a condition characterized by total or near-total paralysis of voluntary movements with largely preserved awareness, cognitive and somatosensory functions (Smith & Delargy, 2005; Bauer et al., 1979). Clinically, LIS is defined by five criteria: sustained opening of the eyes with preserved vertical eye movement, preserved basic cognitive capacities, severe hypophonia or anarthria, quadriplegia or quadriparesis, and vertical eye movement or blinking as primary mode of communication (ACRM, 1995). The extent of motor and verbal impairment leads to different subclasses of LIS, ranging from partial to complete (patients without control of vertical eye movements nor blinking, hence unable to communicate; Bauer et al., 1979).

LIS etiology is very diverse. It can be caused by brain stroke, traumatic brain injury, and neurodegenerative disorders like amyotrophic lateral sclerosis and severe forms of Guillain-Barré syndrome (Haig et al., 1987; Patterson & Grabois, 1986). The anatomic region typically affected is the ventral pons, although extensive bilateral damage of the corticobulbar and corticospinal tracts in the cerebral peduncles may also be responsible for the syndrome (Beaudoin & De Serres, 2010; Smith & Delargy, 2005). Its prevalence rate is not properly documented but research from stroke survivors shows locked-in patients are thought to represent less than 1% of this population[1], though diagnosis is not always reported (some individuals die during the acute phase while others recover before diagnosis is obtained; Beaudoin & De Serres, 2010). Regardless of

[1] About 795,000 people have a stroke every year (AHA, 2016)

its etiology, more than 80% of locked-in patients are still alive after ten years of the syndrome's onset (Doble et al., 2003; Casanova et al., 2003; Schnakers et al., 2008).

Different batteries of neuropsychological tests (short- and long-term memory, attention, executive function, language and verbal intelligence tests) suggest that locked-in patients can recover intact cognitive levels in cases of pure brain-stem lesions (after the acute period). Additional injuries beyond the brain-stem (specially in cortical areas) are most likely responsible for associated cognitive deficits (Schnakers et al., 2008; Rousseaux et al., 2009; New & Thomas, 2005).

Given the survival rate, level of awareness and cognitive functions in this population, providing access to effective communication and mobility are the primary goals of long-term LIS treatment (Wolpaw et al., 2002; Kübler et al., 2005; Bruno et al., 2011).

2. Brain Machine Interfaces (BMI)

Brain-machine interfaces (BMI) aim to restore communication and mobility in the locked-in patient by recognizing changes in brain signals and translating (decoding) them into commands for operating computers, robots and electronic equipment such as spelling devices (Birbaumer, 2006b; Friehs et al., 2004; Brumberg et al., 2011; Bacher et al., 2015). However, currently available technologies do not meet the communication needs of those suffering LIS. This is due to technical and physiological factors that affect BMI performance such as the paradigm type, the decoded signals, and the users' cognitive capabilities (Blain-Moraes et al., 2012; Brumberg & Guenther, 2010).

2.1 BMI considerations

Extensive research has examined multiple signals to decode intentions, from invasive spike recordings and cortical local field potentials (LFPs; Bansal et al., 2012; Santhanam et al., 2006) to non-invasive electroencephalography (EEG) signals (Chaudhary et al., 2016) and magnetoencephalography activity (MEG; Mellinger et al., 2007). Also, near infrared spectroscopy (NIRS; Luu & Chau, 2009) and functional Magnetic Resonance Imaging (fMRI) blood-oxygen-level dependent (BOLD) responses (Naci et al., 2012) have been explored in BMI, yet several barriers remain for finding proper communication channels for translating voluntary commands. Invasive technologies suggest promising results (Bacher et al., 2015; Leuthardt et al., 2006; Krusienski & Shih, 2011) but they remain unappealing to most patients due to the risks of surgery and infection (Birbaumer, 2006a). EEG-based BMI are widely explored for communication but are typically cognitively demanding, require training and sustained attention, intact active vision (which many locked-in patients lack), and conscious modulation of brain signals (Wolpaw et al., 2002); furthermore they often suffer from poor performance in locked-in patients (Brumberg & Guenther, 2010). NIRS BMI are very slow to operate due to the slow hemodynamic response (~10 seconds), and are very sensitive to artifacts, and to the presence of hair obstructing the recording sites (Sitaram et al., 2007). Finally, MEG and fMRI-based technologies are expensive, complex and currently not suitable for daily use by the locked-in patients.

These barriers go beyond technical issues; extinction of goal-oriented and output-directed behaviors following LIS onset can affect the cognitive skills that most BMI paradigms require. Such skills include sustaining attention, learning a new mental task, or modulating cortical and peripheral signals (Birbaumer, 2006b; Kübler & Birbaumer, 2008; Schnakers et al., 2008). Current brain-machine interfaces largely neglect the cognitive limitations of LIS patients.

Moreover, only a small portion of BMI research has been done on patients (Kübler & Birbaumer, 2008; Sellers & Donchin, 2006; Stoll et al., 2013; Brumberg et al., 2011; Kübler et al., 2013; Bacher et al., 2015). Although new efforts are being made to move the technology to home and hospital settings, another major obstacle still remains: brain-machine interfaces do not work for the population as a whole (either healthy or locked-in). Even for the most common BMI paradigms, a percentage of users cannot successfully[2] control the system; a phenomena unfortunately known as BMI illiteracy (also called brain-computer-illiteracy or BCI illiteracy, Allison & Neuper, 2010); although it should be noted that blaming the subject's instead of the algorithms is a big mistake.

Different approaches have emerged to overcome such hurdles, in particular for improving decoding performance. These consist of developing hybrid systems to include other signals or selection modalities (Ahani et al., 2014; Bacher et al., 2015; Millán et al., 2010), improving artifact rejection (such as eye blinks or muscle artifacts (Parra et al., 2005; Parra et al., 2008)), increasing the signal-to-noise-ratio (SNR; McFarland et al., 1997; Rivet et al., 2009), developing better and more robust decoders via adaptive decoding (Gürel & Mehring, 2012; Blumberg et al., 2007; Shenoy et al., 2006), and

---

[2] With *successful* being a subjective measure of performance that is, most of the time, established by each BMI study

incorporating decoder error detection into the classification pipeline (Wolpaw et al., 1998; Spüler et al., 2012). In spite of all these efforts, locked-in patients still face major challenges for controlling apps and systems through BMI.

In this thesis we propose and implement a paradigm that uses error-related potentials for direct binary decoding. By focusing on error monitoring, this paradigm takes into account some of the possible cognitive limitations of LIS patients since error detection is natural to humans, does not require specific training or high levels of attention.

2.2 BMI and adaptive decoding

Adaptive decoding (AD) covers any approach allowing a BMI system to comply with the non-stationarities (changes) of its input data. Amongst these methods are modifying i) the raw signals, ii) the way signals are preprocessed, iv) the features extracted, and v) the decoding algorithm employed.

We can classify AD as supervised and unsupervised. In supervised AD we must know the ground truth for each trial. This information is needed for retraining or recalibration of the system (between or within sessions, Blumberg et al., 2007), or for feature extraction adaptation (McFarland et al., 2011; Shenoy et al., 2006). In the unsupervised case, no ground truth about the trials is required. It can recalibrate the features or the decoder using new data "labels" obtained from unsupervised clustering algorithms (Vidaurre et al., 2011) or from detecting decoder errors (Spüler et al., 2012). It can also retune the decoder using teaching signals and reinforcement learning (Blumberg et al., 2007; Gürel & Mehring, 2012; Sanchez et al., 2009), or apply co-adaptation of the decoder by modifying parameters (such as noise covariance matrices) that can be updated without knowledge of the trial labels (Dangi et al., 2013). In Chapter II of this thesis, we will explore detecting decoder errors using error-related LFP.

2.3 BMI and human-robot interaction (HRI)

BMI can not only be used for communication but also to ambulate and perform daily tasks via the control of an external agent such as a robot. Human–robot interaction (HRI) is the field of study dedicated to understanding, designing, and evaluating robotic systems for use by or with humans (Kosuge & Hirata, 2004). Interaction is tightly related to communication (Goodrich & Schultz, 2007) and brain activity, including EEG error-related potentials (ErrP), have already been assessed as a control channel for HRI: for example, flying a quadcopter to explore the surroundings using the sensory-motor rhythm (SMR, LaFleur et al., 2013), semi-autonomously navigating a wheelchair using ErrP (Perrin et al., 2010), and controlling a robotic arm for target selection employing ErrP as reward signals (Iturrate et al., 2010; Bhattacharyya et al., 2014). Nevertheless, in these studies ErrP for binary-choice selection were studied offline. Current technologies using ErrP for decision making during HRI do not provide online closed-loop control of the system. In this thesis we study the case of closed-loop control of a human-robot interaction system based entirely on online single-trial detection of ErrP signals.

3. Electrophysiological signatures of error processing

Error processing plays an important role in performance monitoring by overseeing goal-oriented behavior. Electrophysiological signatures of error processing have been documented non-invasively using EEG signals (Falkenstein et al., 1991; Gehring et al., 1993), and invasively using electrocorticography activity (ECoG; Milekovic et al., 2012), local field potentials (Emeric et al., 2010; Geng et al., 2013), and spike data (Amiez et al., 2005). During brain-machine control, where decoders can make errors, these error signals have being studied in EEG signals (Ferrez & Millan, 2005; Spüler et al., 2012; Llera et al., 2011) and in subcortical spike data (Prins et al., 2013).

3.1 Local field potentials (LFP)

LFP are low-frequency (usually below 300 Hz) electrical signals recorded from extracellular electrodes placed (invasively) within or on the surface of the brain (Pesaran, 2009). LFP reflect the summation of activity of synaptic potentials from neurons in a region of tissue, approximately within 250 μm of the recording electrode (Katzner et al., 2009).

Emeric et al. (2008) found that LFP from the anterior cingulate cortex (ACC) during a saccade stop signal task (go/no-go task) in macaque monkeys signaled error and reinforcement in a manner consistent to the EEG error-related activity found by Falkenstein et al. (1991) and Gehring et al. (1993). Similar results have being found in the supplementary eye fields (SEF) of macaque monkeys (Emeric et al., 2010). Likewise, Geng et al. (2013) found LFP from the ventral striatum in marmoset monkeys encoding error-related activity during a binary selection task involving a robotic arm. These results suggest a relationship between error-related LFP (eLFP) and error-related EEG signals. In spite of these advancements, error-related activity from LFP has not been explored nor studied during BMI control for decoder error detection.

3.2 Electroencephalography (EEG)

Electroencephalography (EEG) is an electrophysiological technique that measures the electrical activity of the brain (usually employing electrodes placed on the scalp and recording activity at frequencies below 100 Hz). It represents the activity of synchronized populations of neurons and is affected by the whole volume conduction of the head (Regan, 1989). EEG signals contain different rhythmic oscillations that are influenced by several external and internal factors such as the level of wakefulness, disease, or external stimuli. The electrical response signals of the brain to a specific stimulus are called evoked-potentials (or event-related potentials; Guyton & Hall, 1990).

3.3 Error-related Potentials (ErrP)

Error-related potentials (ErrP) are EEG event-related potentials that result from being exposed to an incorrect outcome during a goal-oriented task (Chavarriaga et al., 2014). Error awareness is not necessary for the presence of ErrP; evidence suggests that ErrP can be present during the commission of an error even when the subject is not consciously aware the error has occurred (Ullsperger et al., 2014).

3.3.1 Anatomy of the ErrP signal. Error-related potentials are commonly presented in the literature as the difference between the averaged trace of the incorrect trials minus the averaged trace of the correct trials (also known as the *error difference* trace, *ErrDiff*). From this signal, characteristic deflections seem to have functional significance (Nieuwenhuis et al., 2001). Two main components of the error difference trace have been described, the error-related negativity or error negativity (Ne), and the error positivity

(Pe). The Ne is characterized by a fronto-central negative deflection reaching its peak from 50 to 100 ms after the response or feedback onset time, and seems to be independent of the stimulus and effector modality (Falkenstein et al., 2000). The Ne is followed by a centro-parietal positive deflection (Pe) that peaks between 200 to 400 ms after the error. Evidence suggests that the Ne is the result of an error-detection mechanism, the comparison process between expected goal and outcome (Hewig et al., 2011); and (although with less certainty) that the Pe component reflects conscious error processing or post-error adjustment of response strategies and the outcome level of confidence (Falkenstein et al., 2000; Ullsperger et al., 2014).

The Ne component of the ErrP is usually measured as the most negative peak (at Fz, FCz, or Cz) in the time window 0–160 ms after the feedback onset time. The Pe is usually measured as the most positive peak at Pz in the time window of 200–500 ms after the feedback, or, if no clear peak is visible, as the mean amplitude in that window (Falkenstein et al., 2000). Although the Ne and Pe are generalized characteristics of the ErrP signal, its amplitude and timing varies depending on the task, presence or not of feedback, and the type of agent performing the action making the mistake (see more in the next section, Types of ErrP).

3.3.2 Types of ErrP. Four types of ErrP, elicited by incorrect responses, have been identified and characterized in EEG data. Differences in the anatomy of their waveforms, their peaks and toughs, are related to elements intrinsic to the task, such as the required motor actions, the presence of feedback, and the type of agent performing the action that led to the error. These ErrP are called response ErrP (Carter, 1998; Falkenstein et al., 1991; Gehring et al., 1993), feedback ErrP (Holroyd & Coles, 2002; Miltner et al., 1997), observation ErrP (Iturrate et al., 2010; van Schie et al., 2004), and interaction ErrP (Ferrez & Millan, 2005). Observation ErrP occur when an external agent (such as another person or a robot) makes a mistake, without the influence of the observer. Interaction ErrP occur during brain-machine interaction, when the action does not match the user's intention due to a decoding error (with the decoder being driven by the user's brain activity). The characteristics of the observation and interaction ErrP traces are of special interest to us since these are the signals we will focus on. The interaction ErrP has a sharp negative peak (occurring 270ms after the feedback onset), followed by a positive peak (between 350 and 450ms after the feedback), and by a later negative peak (around 550ms after the feedback; Ferrez & Millan, 2005; Buttfield et al., 2006).

Moreover, in these ErrP the peak amplitude has been correlated to the degree of error occurring during the BMI task (Iturrate et al., 2010; Iturrate et al., 2014).

3.3.3 ErrP and BMI. ErrP have being employed in BMI for detecting decoder errors (Wolpaw et al., 1998) and performing adaptive decoding (Spüler et al., 2012; Gürel & Mehring, 2012). To our knowledge, they have never been used for direct binary control. Also, ErrP signals have been found in paradigms where errors occur in 50% of the total trials; this property differentiates these signals from mismatch negativity potentials (MMN; which require an infrequent stimulus, as in the oddball paradigm, Buttfield et al., 2006; Chavarriaga & Millan, 2010; Ferrez & del R Millan, 2008) and suggests that ErrP can be used for direct binary selection. In Chapter III of this thesis, we will present a closed-loop BMI that employs observation and interaction ErrP during human-robot interaction.

4. Neural correlates of error processing

Performance monitoring is essential to regulate goal-oriented behavior since it allows adaptation to unexpected changes (Scheffers & Coles, 2000). Error processing is part of the performance monitoring system, and is believed to be involved in boosting attention to the incorrect actions and enhancing available resources in executive functions with the goal of avoiding similar errors in the future (O'Connell et al., 2007; Vocat et al., 2008). These changes in attention seem to modify the autonomic nervous system's (ANS) arousal state and, as a consequence, all the peripheral signals controlled by the ANS (Hajcak et al., 2003). Error-related activity, measured both invasively and non-invasively, has been used for detecting several anatomical structures involved in error processing. Based on analysis of MEG, EEG and fMRI data from humans, as well as spike and LFP data from non-human primates, the cortical areas commonly linked to error-related activity cover mainly the posterior medial frontal cortex (pMFC), the anterior cingulate cortex (ACC), dorsolateral prefrontal cortex (dlPFC), and the pre-supplementary motor area (pre-SMA, Ullsperger et al., 2014; Holroyd & Coles, 2002). In one study of ECoG in humans with epilepsy, the motor and somatosensory areas also revealed signatures of error-related activity during BMI control (Milekovic et al., 2012). Overall, this error-related activity is believed to encode evaluative signals about the state and outcomes of actions and integrate this information to signal the necessity, type, and magnitude of adjustments needed to correct performance (Ullsperger et al., 2014).

5. Summary of Dissertation: Motivation and Approach

The goal of this dissertation is to evaluate the use of decoder error detection for adaptive decoding and volitional control, for error correction using eLFP, and for binary-choice selection using observation and interaction ErrP. This project contributes to the fields of performance monitoring, BMI, augmentative and alternative communication (AAC), and HRI. Taken together these results highlight the versatility of error-related potentials for both adaptive decoding and binary-choice selection.

5.1 Chapter II: Real-Time Adaptive Decoding in an Invasive Saccade BMI

Chapter II focuses on employing eLFP for implementing real-time single-trial decoder error detection for adaptive decoding. We initially identify characteristic signatures present in the eLFP that reflect decoder errors during BMI control. Second, employing the traits found in error-related LFP, we present offline and online results of a real-time decoder error detection system.

Analysis of data taken from two macaque monkeys performing a delayed-saccade task controlled by a BMI indicate that eLFP have characteristic features related to the magnitude and valence of decoder errors (errors made by the saccade BMI), as have been reported in non-human primates during non-BMI tasks (Emeric et al., 2010) as well as in humans in EEG-based BMI (Chavarriaga & Millan, 2010). As described for EEG signals (Ferrez & del R Millan, 2008) and LFP (Alexander & Brown, 2011; Scheffers & Coles, 2000), the amplitude of the eLFP is inversely proportional to the error likelihood (the more frequent the error, the lower the amplitude). Also, the eLFP peak amplitude is correlated to the level of discrepancy of the errors (the farther the incorrect target to the correct one, the larger its amplitude). These features are present in all three areas (SEF, PFC and FEF) but with variations specific to each implant. Finally, the stability of the signals over time (months) suggests eLFP to be better suited for long term BMI than approaches based on spikes.

For the real-time single-trial decoder error detection system, we developed and implemented a classifier working in parallel with the saccade BMI. Offline analysis of the eLFPs revealed the type of training needed for the online sessions as well as the best features for each monkey, reaching performance values above 95% for both monkeys. Real-time online classification provided decoding performance of 87% across monkeys, 86% for monkey C and 88% for monkey J. These results support the possibility of using error-related activity for adaptive decoding in invasive BMI and shed light on the neural correlates of decoder errors during BMI control.

5.2 Chapter III: Error-related potentials for direct volitional control and human-robot interaction

In Chapter III we continue exploring the advantages of employing error-related potentials for BMI but we focus on its application for direct volitional control. As mentioned earlier, error-related potentials can be used not only for recalibrations of the decoder (adaptive decoding) but for direct control, especially in binary-choice paradigms. Chapter III focuses on the use of ErrP (from humans) and decoder error detection for controlling a robot performing a closed-loop binary object selection task. Although technically more challenging to implement, we use a robotic arm to increase the engagement of the participants in the task. Due to the nature of our paradigm, signals from our participants reveal observation and interaction ErrP during online closed-loop sessions.

We collected data during open- and closed-loop sessions, and performed single-trial error detection both offline and online. Analysis of observation and interaction ErrP reveal features, timing and co-localization agreeing with the literature (Ferrez & Millan, 2005; Buttfield et al., 2006). Offline analysis of decoder errors involving both types of ErrP suggest interaction ErrP are easier to classify than observation ErrP, with an average performance across subjects of 69% for observation ErrP and 83% for interaction ErrP. Moreover, from the pool of subjects recruited we found that all of them present the ErrP (except in a subject performing meditation), a critical element when considering other paradigms and BCI illiteracy.

Although preliminary, the results of these analyses suggest that direct volitional control using only ErrP may be possible, and that decoding performance can be improved if the closed-loop system performs decoding of both types of ErrP.

5.3 Chapter IV: Conclusions

We conclude this dissertation by compiling all the findings, discussing their implications for BMI, ACC and HRI, and proposing future directions, especially exploring the use of ErrP for communication.

CHAPTER II: REAL-TIME ERROR DETECTION IN AN INVASIVE SACCADE BMI

1. Introduction

For almost two decades, brain-machine interfaces (BMI) have promised a natural way to augment human capabilities, both for the healthy and the disabled, by means of distinguishing different patterns in brain activity (Millán et al., 2010). This promise, however, has been difficult to bring to fruition. The technology has always endowed those needing it with hope, but multiple technical and practical issues have made the BMI race a long one (Birbaumer, 2006a). For both invasive and non-invasive BMI these challenges cover its different functional blocks: signal acquisition and processing, control algorithm, device output, and operating protocol (Leuthardt et al., 2006). Multiple solutions have been proposed to improve BMI classification performance, including adaptive decoding and decoder error detection (Gürel & Mehring, 2012; Blumberg et al., 2007; Shenoy et al., 2006; Spüler et al., 2012).

In this chapter we investigate the use of cortical error-related LFP (eLFP) in a real-time system for detection of single-trial decoder errors produced while a non-human primate (NHP) controls a closed-loop saccade BMI. Initially, the eLFP are described; then offline analyses of the decoder error detection algorithm are performed, followed by real-time test of the decoder error detection (DED) system during saccade BMI control.

To our knowledge, the use of eLFP during BMI control has never been described in the literature, nor used in a closed-loop system for detecting single-trial decoder errors. LFP from the striatum in marmoset monkeys have been explored offline (Geng et al., 2013), but not in closed-loop settings. To bridge what we know on error-related activity both in invasive and non-invasive BMI, first we focus on thoroughly characterizing the eLFP, both in the time and frequency domains. The key aspects to evaluate are differences between correct and incorrect decoding trials under different analysis conditions: target location, distance to true target (Iturrate et al., 2010), and previous trial outcome (Cavanagh et al., 2010). Spectral analyses of the eLFP explore the influence of decoder errors on specific frequency bands (Chavarriaga et al., 2014), and cross-frequency amplitude-amplitude coupling analysis of the signals aim to detect frequency co-modulation (Masimore et al., 2004).

In this chapter, offline simulations of the DED algorithm cover a series of iterative processes searching for the optimal parameters and training dataset (via 10-fold cross-validation) to be used during the real-time sessions. In these real-time sessions, a trained and optimized DED classifier is in charge of detecting single-trial saccade BMI decoding errors and outputting a label that could be used by an adaptive algorithm. Finally, with the results of the real-time DED BMI, the effect of implant laterality on detecting saccade BMI errors is explored. The eLFP characterization, training and testing used data from two macaque monkeys performing a delayed-saccade task.

The novelty of this approach is that we detect decoder errors using cortical LFPs in a real-time closed-loop system. In our current approach we do not present the feedback of the DED BMI classification to the monkey, just to the saccade BMI algorithm.

Most of the emphasis of this chapter is in the Results section, specifically the eLFP characterization. The Methods section presents the structure of the data and how it is regrouped and analyzed (based on different conditions), the different statistical tests run between the decoder outcome conditions, as well as the techniques for selecting the decoder that were implemented in the testing sessions. In the Results section we describe the properties of eLFP in the time and frequency domains, and we provide the results of offline simulations and real-time DED BMI tests. Finally, we discuss the contributions of this work to the literature and the future directions of our research.

2. Methods

Adaptive decoding covers a broad spectrum of techniques aiming to improve BMI performance. One focuses on online adaptation of the decoder based on error-related potentials and unsupervised learning. In this case, the decoder is retrained (online) using data labeled in an unsupervised manner. During each trial, after the decoder makes a prediction, an error-related potentials classifier labels the current trial as correct or incorrect. If correct, the data and the label from the trial are appended to the training dataset and the decoder is retrained (online). If the error-related potentials classifier has high accuracy, the overall decoder’s performance has been shown to increase due to this type of online adaptation (Spüler et al., 2012). In this section we present the methods employed to detect (offline and in real-time) eLFP during a saccade BMI task. We start by presenting the techniques employed to identify characteristic error signatures in the eLFP, followed by analysis of how these characteristics reflect decoder errors during BMI control. Finally we use these traits in a real-time single-trial decoder error detection system. The methods presented here, including the system implemented for real-time decoder error detection, can be used not only for analyzing error-related potentials in LFP but also in electrocorticography (ECoG) and electroencephalography (EEG) data.

2.1 Non-human primates and implants

Data were recorded from two adult male macaque monkeys (monkey C, *Macaca fascicularis*, 9kg; monkey J, *Macaca mulatta*, 11kg) at the Miller lab in the Picower Institute for Learning and Memory at Massachusetts Institute of Technology (MIT). Handling of the animals followed all the guidelines approved by the MIT Committee on Animal Care.

Both monkeys were proficient in a memory-guided saccade task before being implanted, unilaterally, with three 32-channel microelectrode arrays (Blackrock Microsystems, Salt Lake City, UT). Each array consists of 32 microelectrodes spaced 400 μm apart and 1 mm in length. Implant locations included the dorsolateral prefrontal cortex (dlPFC), frontal eye field (FEF), and supplementary eye field (SEF). Arrays in monkey C were implanted in the left hemisphere; those in monkey J were implanted in the right hemisphere. The location of the implants for both monkeys is presented in Figure II.1.

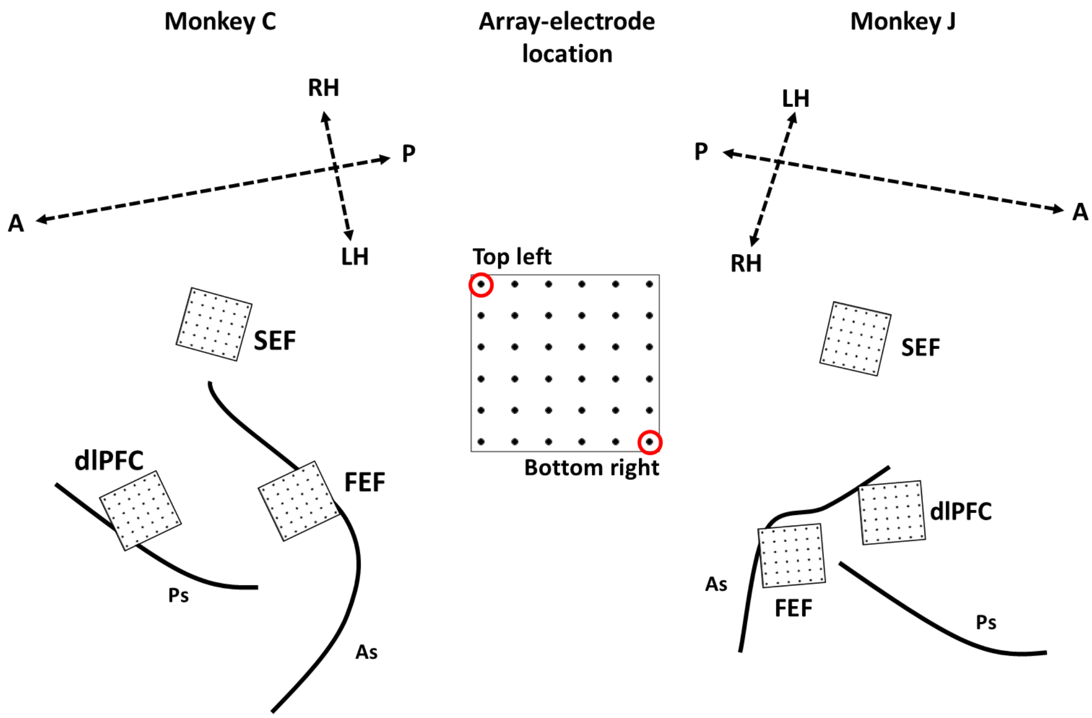


Figure II.1: Implant locations of microarrays for both monkeys.

Implant locations for both monkeys. Squares represent 32-channel microelectrode arrays with electrodes ordered from top to bottom and left to right. A: anterior; P: posterior; RH: right hemisphere; LH: left hemisphere; Ps: principal sulcus; As: anterior sulcus; dlPFC: dorsolateral prefrontal cortex; FEF: frontal eye fields: SEF: supplementary eye fields. Red circles present the location of the array's top left and bottom right electrodes.

2.2 Data recording

Local-field potentials (96 channels) were sampled at 5 kHz (Cerebus BlackRock Microsystems) and extracted via two different pipelines, one for the saccade BMI, and another for the decoder error detection (DED) BMI. LFP for the saccade BMI were low-pass filtered with a third-order Butterworth filter with a pass-band frequency of 80-500 Hz and recorded at 1 kHz. For the real-time DED BMI tests, LFP were band-pass filtered with a second-order Butterworth filter with a pass-band frequency of 1-10 Hz. Data collection and implementation of the saccade and DED BMI were performed in Matlab (Mathworks) with the PsychToolBox toolbox (available at http://psychtoolbox.org/, Brainard, 1997). Eye movements were monitored using an infrared eye tracking system (EyeLink 1000, SR Research) at a sampling rate of 1 kHz.

2.3 BMI paradigms

In this section we describe two different BMI. First, a saccade BMI provided data related to decoder errors. Second, a decoder error detection BMI explored the use of error-related LFP for detecting, in real-time, the decoder errors made by the saccade BMI.

2.3.1 Saccade BMI. The BMI paradigm consisted of simultaneously recording neural and behavioral data while the monkeys (who sat in a primate chair, 30 cm from a cathode ray tube screen, with the head fixed) performed a delayed saccade task, a modified version of the memory-guided saccade task. Initial sessions were used for extensive offline analysis to find the best parameters and decoder for the saccade BMI. The chosen features were the mean power of the LFP 80–500 Hz frequency band, across the entire delay period, for each channel. Their target direction selectivity was computed using one-way ANOVA and only the features with p-values smaller than 0.05 were used as input to a linear regression classifier (Jia et al. 2017). These signals and decoder were chosen because they provided the highest decoding accuracy. For implementing the saccade BMI, each session was divided into two blocks, training (or eye-controlled) and decoding (or BMI-controlled). On each recording day, the data from the training trials were used to train the classifier, which was then tested in the decoding trials.

During each trial (Figure II.2, blocks on the left of the red dashed line), monkeys were required to fixate on a dot in the center of the screen and were presented (for 350ms) with one of six potential saccade targets (at 12.5-degree eccentricity, and evenly spaced every 60°). The monkey had to hold this location in working memory over a delay period of 750ms while maintaining fixation at the center on the screen. At the end of the delay period the fixation point was removed to cue a monkey saccade (on a training trial), or a decoded BMI saccade (on a BMI-controlled trial). Any trials with broken fixation or premature saccades were aborted.

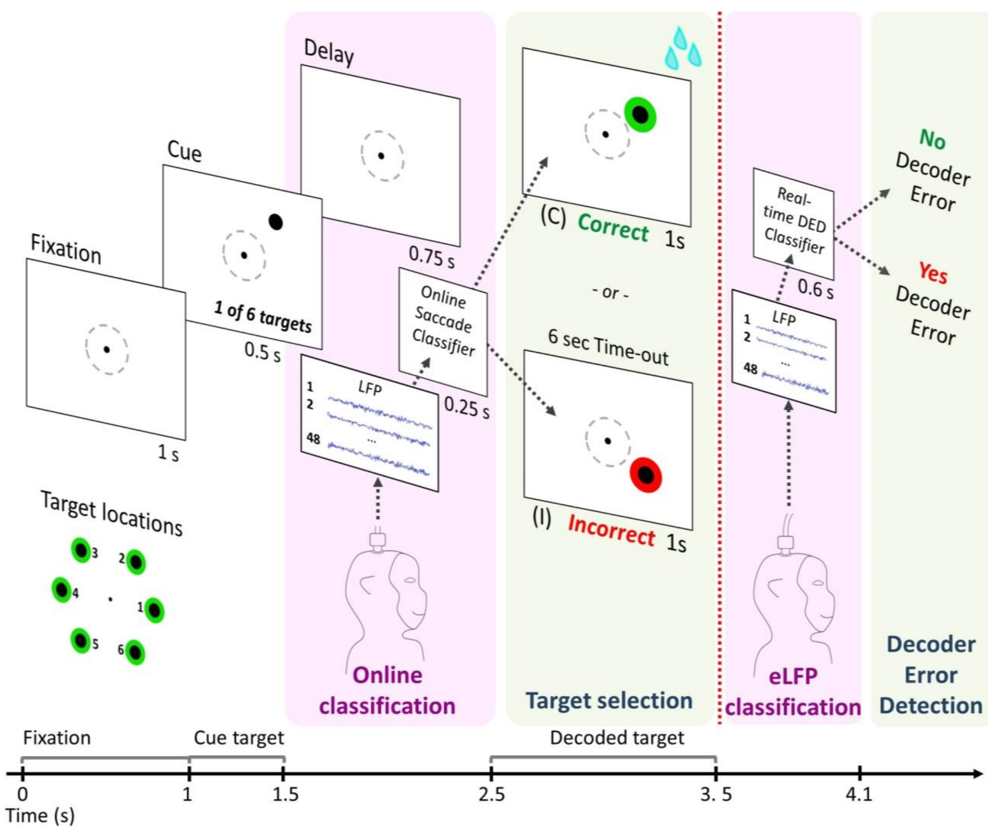


Figure II.2: Brain-controlled delayed-saccade task with real-time decoder error detection

A session of the delayed-saccade task controlled by the saccade BMI was divided in two blocks: the training and BMI-controlled blocks. In the training block, monkeys performed saccades to earn a reward. In the BMI-controlled block a decoder replaced the motor responses by predicting a saccade and presenting it on the screen. During both blocks, each trial was initiated by central fixation. Then one of six potential saccade targets was randomly cued. Monkeys were required to hold the target location in working memory for a delay period. Upon the end of this period, monkeys were required to saccade (on a training trial), or the BMI predicted the saccade (on a BMI-controlled trial). Positive or negative reinforcement —liquid reward and a green target, or a 3-second timeout for the next trial and a red target, respectively— was delivered conditional on whether the saccade/predicted location matched the instructed one. For the BMI-controlled trials the intended target was decoded from LFP activity recorded during the delay period. Decoder error detection was performed after the presenting the predicted location using LFP activity (blocks on the right of the red dashed line).

During the training trials, motor responses from the monkey (saccades) determined the trial outcome. On the decoding trials the saccade BMI controlled the task. The intended target was decoded from neural activity during the delay period and then a cursor was presented at the decoder-predicted location, replacing the overt motor response. Positive or negative reinforcement —liquid reward and a green target, or a 3-second timeout for the next trial and a red target, respectively— was delivered conditional on whether the saccade/decoded location matched the instructed one. The positive or negative reinforcement was use as a feedback signal of the trial outcome. The liquid reward was delivered by a solenoid that was activated three times by consecutive 40ms pulses, 10ms apart (spanning a total of 100ms).

During each recording session testing the saccade BMI, monkey C executed 600 training trials and at least 700 decoding trials. Subsequent offline analysis during the BMI-controlled sessions revealed that more than 300 training trials did not significantly increased the decoder performance, hence monkey J executed 300 training trials and at least 1000 decoding trials.

2.3.2 Real-time decoder error detection BMI for adaptive decoding. The saccade BMI made classification errors, particularly in decoding targets placed ipsilateral to the hemispheres where the arrays were implanted.

In order to improve decoder performance, a real-time decoder error detection (DED) module (BMI for adaptive decoding) was added to the saccade BMI. The DED BMI (Figure II.2, blocks on the right of the red dashed line) was in charge of detecting single-trial decoded errors by classifying the decoded target as correct or incorrect (using LFP activity recorded after the feedback presentation). This information was meant to be used by the saccade BMI to adapt its decoded target (which should be tested in the future using and adaptive algorithm). Extensive offline analysis (Section 2.4) determined the best pre-processing, time windows, and features for reliably detecting decoder errors. These features were used to train a linear regression classifier (prior to the testing session) that remained fixed throughout the entire session. The real-time DED BMI was tested three weeks for monkey C, and two weeks for monkey J. More details on the implementation of the real-time DED BMI can be found in Section 2.5.

In the following sections a naming convention will be used to differentiate the two BMI. Any trial or procedure that is part of the DED BMI, and not the saccade BMI, will be followed by the suffix (err). Specifically, trials that were correctly and incorrectly decoded by DED BMI will be referred as correct(err) and incorrect(err) trials. The DED decoder will be named as decoder(err) for simplicity. A correct(err) trial is a trial on which the DED BMI classifier behaved as it should: either an incorrect trial that the DED BMI classified(err) as an **error** (a saccade BMI error correctly detected), or a correct trial that was decoded(err) by the DED BMI as a **no error**. Trials correctly and incorrectly decoded by the saccade BMI will not have a suffix appended.

2.4 Offline analysis

Our offline analysis aimed to develop a real-time system capable of detecting single-trial decoder errors. In this section we describe the methods applied to study the temporal and time-frequency characteristics of the eLFP, and the approach taken for optimizing the parameters used during offline decoder error detection. All statistical tests were performed with balanced data (when comparing the correct and incorrect classes, using from each class the same number of trials) from 1000 bootstrap iterations (with replacement).

2.4.1. Temporal analysis of error-related local field potentials (eLFP). Only data collected during the decoding block of the saccade BMI were used for these analyses. LFP were filtered (offline) using a fourth-order Butterworth filter with a pass-band frequency of 1-10Hz. Epochs from the filtered signals were extracted using as criteria the feedback onset time. An epoch spanned 600ms pre-feedback to 600ms post-feedback onset time.

Artifact rejection: artifacts in the recordings were removed using a semi-heuristic thresholding method. Using the 1-10Hz epochs, the mean and standard deviation (S.D.) across trials (for each channel and session) were computed in the interval 0-600ms post-feedback onset (for the correct and incorrect trials independently). Any trial with amplitudes outside the range of the mean (of each channel) ±3S.D. was labeled as 'bad' (for the specific channel), and if more than 15 channels in the same trial had the 'bad' label, that trial was removed from the analysis. The trials labeled as 'bad' were also removed from the data (0-200Hz) used to compute the spectrograms.

Decoder outcome condition: based on the saccade BMI decoding outcome, epochs were grouped into correct and incorrect.

Target location condition: the saccade BMI made classification errors in all of the 6 possible target locations. Trials were also regrouped based on their true target location, resulting in correct and incorrect epochs for each one of the targets.

Averages of all the epochs for the 1) decoding outcome, and 2) target location conditions were computed and plotted. Error difference traces (ErrDiff) were obtained for each channel as the difference between the mean incorrect and mean correct traces (Ferrez & Millan, 2005). A two sample T-test was applied to the time traces to compute the variance explained by the decoding outcome, at each target, channel and time sample (using a p-value < 0.01 criteria for significance).

Distance to true target condition: with six targets evenly distributed, we defined the distance to true target criteria to evaluate how the amount of error affected the amplitude of the eLFP (see Figure II.3). The distance to target reflects how far apart the saccade BMI decoded target was from the true target (in circular coordinates). It regrouped data into 4 subsets: dist0 (when the target was correctly decoded, 0º apart), dist1 (when the correct target was 1 target away on either side, 60º apart), dist2 (when it was two targets away, 120º apart), and dist3 (on the opposite side, 180º apart). For each session, incorrect traces were regrouped based on their distance to true target, and averaged (across trials). The channels and time samples that contained significant information about the amount of error were determined using a one-way ANOVA ($p < 0.05$). ANOVA results only inform that at least one of the distances to true target subsets is significantly different. To assess if all subsets are different from each other, we performed two sample T-tests ($p < 0.01$) across all pairs of distance to error (dist1-dist2, dist2-dist3, dist1-dist3).

Previous trial outcome condition: incorrect outcomes are known to lead to a slower reaction time on the subsequent trial. To evaluate this effect, epochs were regrouped into those incorrect after a correct trial (corr1incorr2) and incorrect after an incorrect trial (incorr1incorr2). Their averages were obtained and significant channels and time samples were identified using a two sample T-test ($p < 0.05$).

Figure II.3. Distance to true target of a decoded trial.

The distance to true target is the amount of error the decoded targets represents. Due to the circular nature of the target location only 4 different distance to true target values can be computed between the correct target and the decoded target: 0 (correct), 1 (60º apart, 1 target away), 2 (120º apart, 2 targets away), 3 (180º apart, 3 targets away).

2.4.2. Spectral and time frequency analysis of eLFP. Unlike in the temporal analysis section, spectral and time-frequency content of the LFP (for both correct and incorrect trials) were computed for all channels using data band-pass filtered in the frequency band

[0-200]Hz via the multi-taper method (Bokil et al., 2010), using the Chronux toolbox for Matlab (available online from P. Mitra, http://chronux.org/). Its parameters were: 2 tapers, half time-bandwidth product of 3, a 0.2 s moving window, and a window step of 0.025 s. Averages of the spectrograms and the frequency bands were computed for correct and incorrect trials as well as the ErrDiff (to evaluate the difference in spectral activity between mean incorrect and mean correct spectrograms).

Six different frequency bands were defined: [1-4]Hz (delta), [4-8]Hz (theta), [8-13]Hz (alpha), [13-30]Hz (beta), [30-80]Hz (gamma), and [80-200]Hz (high gamma). The average power across these bands was computed for each time sample in all correct and incorrect spectrograms. A two sample T-test was run for each band. The average values for all frequency bands were computed for all trials (for each monkey), and the average error difference frequency band was obtained as the difference between the average incorrect and average correct frequency bands. The error difference frequency band was used to evaluate channels and arrays that had larger spectral differences due to the decoder outcome (two sample T-test).

2.4.3. Spectral cross-frequency co-modulation. To evaluate if specific frequency bands were correlated, cross-frequency amplitude-amplitude coupling was used (Masimore et al., 2004). The cross-correlation of the amplitude of the frequency bands was computed for the correct trials, and separately, for the incorrect trials. Also, this was done with data from pre- and post-feedback periods (for each type of trial outcome). The cross-frequency amplitude-amplitude coupling is defined as:

$$\rho_{ij} = \frac{\sum_{t=1}^{N}(S_t(f_i) - \bar{S}(f_i)) * (S_t(f_j) - \bar{S}(f_j))}{(N-1)\sigma_i\sigma_j}$$

where $S_t(f_i)$ is the spectral density at frequency f_i in trial t, $\bar{S}(f_i)$ is the average spectral density magnitude at frequency f_i over all trials, σ_i is the standard deviation of the spectral density at frequency f_i, and N is the total number of trials for the evaluated condition. The range of this metric is -1 (decoupling), 0 (no coupling), and 1 (high coupling). The difference of the post- and pre-feedback period cross-frequency coupling (periodDiff coupling) was computed for each outcome (incorrect and correct) in order to account for the real increases or decreases in coupling after the feedback presentation. The difference between the incorrect and correct periodDiff coupling was computed to assess any changes in frequency coupling due to decoder outcome. The values presented in the Results section are the average of 1000 iterations using bootstrapping (with replacement).

2.4.4. Offline decoder error detection. We determined the best parameters for reliably detecting decoder errors offline following a two-step analysis. First, *cross-validation* was performed for evaluating different parameters of the DED decoder(err). Second, we used *different sessions for training and testing* to establish the appropriate training for the real-time testing sessions. In the *cross-validation* approach we ran, for each session, 10-fold cross-validation of DED models, each model using a different set of parameters. These parameters were: 1) baseline removal, 2) type of arrays, 3) feature extraction function, and 4) feature transformation function. Their possible values were:

Baseline removal: a baseline was computed from each channel and epoch. It was computed as the average of the activity for the 200ms before the feedback onset time. The possible options for this parameter were removing the baseline (1Base) or not (0Base).

Type of arrays: a subset of all available channels could be chosen by means of selecting only some of the arrays. The possible array combinations were: SEF only; PFC only; SEF and PFC; SEF, PFC and FEF; FEF only; FEF and SEF; and FEF and PFC.

Feature extraction function: to provide a fast, real-time, DED BMI, we leveraged previous work that characterized the incorrect traces using peaks and troughs (Falkenstein et al., 2000; Chavarriaga & Millan, 2010). First, we defined 5 data windows, which represented the most common peaks and troughs in the signal (these windows were selected using a semi-heuristic method and extensive visual inspection of the correct and incorrect epochs). The windows were bounded by the post feedback onset times 50-100, 100-150, 150-250, 250-350, and 350-600ms (see Figure II.4). Second, we obtained different feature vectors by applying a set of functions to each of the data windows: mean, median (label in the figures as mean2), and the minimum and maximum (minMax). Thus, considering the five windows, the mean and median functions each output a vector of 5 features per channel. The minimum and maximum function outputs 2 values per data window, and thus a feature vector of 10 features per channel.

Feature transformation function: different transformation functions were applied to the feature vectors. These included: getting the feature’s square root (sqrt), taking its square value (sq), logarithm (log), and subtracting its mean (mean) and z-scoring it (zscore) using all the trials from a session or population of sessions (see section 2.3.1 for the expected number of trials per session). Finally, there was also the option of not performing any feature transformation (none).

All the possible combinations of these parameters were tested in the 10-fold cross-validation pipeline using a linear regression classifier. For each iteration, the output of the classifier was discretized to label the trials as correct or incorrect, and to test its decoding(err) performance. The parameters providing the best DED performance were used in the second part of the analysis pipeline (see next paragraph), and chosen for the real-time decoder error detection BMI. The average and median of the decoder(err) performance was computed and plotted across 65 monkey C sessions, and 9 monkey J sessions.

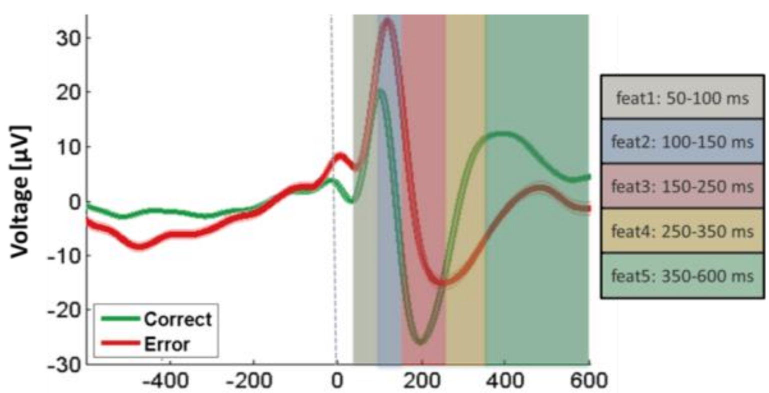


Figure II.4. Data windows used for feature extraction.

A representative channel from SEF in monkey C. Color boxes represent the data windows used to extract the different features from the filtered data, 5 windows in total. These windows bound different peaks and troughs in the signal. The green and red traces represent the average of all correct and incorrect trials, respectively. X axis represents the time to feedback onset in milliseconds, the Y axis the signal amplitude (in µV).

In the second part of the analysis we mimic the training and testing sequence of the real-time DED BMI. A decoder(err) that was going to be tested using data from a specific day was trained with data from sessions prior to that day (usually the latest sessions). We performed this analysis for 65 sessions from monkey C. For monkey J we used the 9 sessions available. Five different rules were used for selecting the training dataset. For monkey C these included choosing: 1) only the last session, 2) the last 5 sessions, 3) the last 10 sessions, and 4) always use the first 33 sessions (the same decoder(err) for all tested sessions, see Figure II.5). Finally, for comparison, we ran the analysis using a fifth rule, 5) 10-fold cross-validation over each session. During these simulations, each one of the testing sessions used a decoders(err) trained with datasets following the 5 rules, and with the parameters optimized in the cross-validation section.

The average performance across all sessions was used to select the first decoder(err) to be used during the real-time DED BMI sessions. Average performance metrics included true positives (decoder errors correctly detected as incorrect), true negatives (correct trials classified as no errors), and overall classification accuracy.

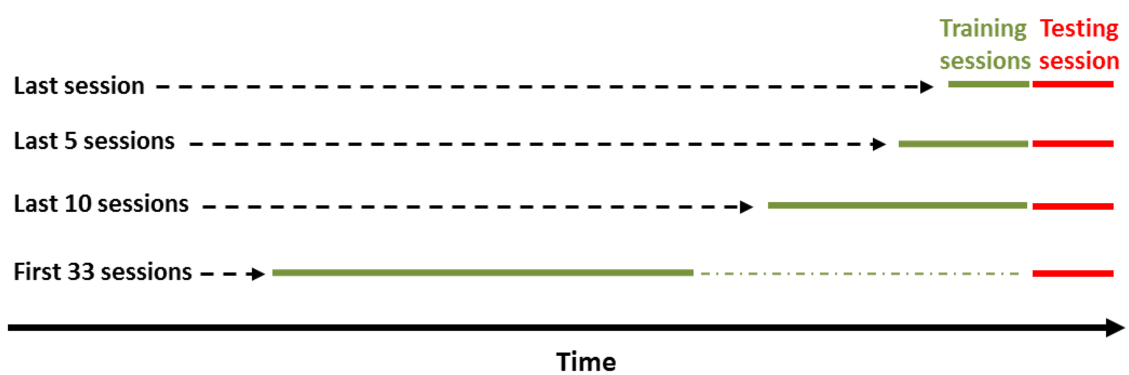


Figure II.5. Training and testing dataset structure mimicking online real-time DED BMI sessions.

Four different training dataset rules were used to determine the number of sessions used for training the decoder(err) during the DED BMI offline testing. Each dataset rule indicated the number of training sessions to be used, all starting from the latest available recording. For a given testing dataset, we could use for training the last session, the last 5, last 10 or the first 33 sessions from a total of 65 sessions.

2.5 Real-time eLFP detection

Real-time decoder error detection was performed only during the decoding block of the saccade BMI (Figure II.2, blocks on the right of the red dashed line). For each trial, following the feedback presentation, LFP in the 1-10Hz pass-band were extracted online in Matlab (Mathworks), and a buffer of 600ms was selected (starting at the feedback onset time). Based on the results of offline analyses, only a subset of channels were chosen for eLFP calculation: SEF channels for monkey C, FEF and PFC channels for monkey J. For each channel, 5 features were computed from the averages of the data windows, bounded by 50-100, 100-150, 150-250, 250-350, 350-600ms post feedback onset time. The 5 features from all the selected channels were concatenated into a single vector of 160 and 320 features, for monkey C and monkey J, respectively. The feature transformation function applied to each monkey's data was chosen after several offline analyses. The feature vectors for monkey C were z-scored (using the mean and standard deviation obtained from the trained trials). Feature vectors from monkey J were not modified.

The trial outcome was predicted with these feature vectors and with the classifier previously trained (with linear regression; see Section 2.5.1). Since linear regression provides a continuous output, it was discretized into the nearest one of the two possible outcomes, 0 and 1 for negative and positive decoder error detection, respectively. The DED BMI module for real-time training and classification was implemented using the PsychToolbox toolbox for Matlab (available at http://psychtoolbox.org/, Brainard, 1997).

2.5.1 Training of the decoder error detector. The DED classifier(err) was trained on data from previous sessions (from the decoding block only). Offline, the data used for training was iterated to select the best classifier (similar to the cross-validation analysis performed offline). Starting with the most recent session, and increasing one by one the number of sessions included in the training dataset (up to 20), a 10-fold cross-validation pipeline was run (using for each monkey the optimized arrays, data windows and features). The only difference between the 20 models was the data (number of sessions) used for cross-validation. From these 20 different models the one with the highest classification performance was chosen, and then a DED classifier(err) was trained using all the sessions from the chosen model. This decoder(err) was saved and stored to be used in the decoding block of the next saccade BMI session.

To test the robustness and generalization of the classifier, the same DED classifier for monkey C was used for the first 5 sessions. For the following sessions the decoder was updated each day. For monkey J, a new classifier was retrained each session. This process of updating the decoder(err) every day was true except on days when the newest data (usually from the day before) were not available for training. In those cases the most recently trained classifier was used instead. For more information on the specific sessions used for testing the decoder(err) see Table II.3 in the appendix.

2.5.2 Real-time testing of the decoder error detection BMI. Testing of the DED BMI was done in the decoding blocks, under saccade BMI control, and using the decoder(err) trained before the sessions (with previous data, and following the parameters optimized;

for more information see Sections 2.5.1 and 2.4.4). After feedback presentation, each trial was classified as correct or incorrect by the DED BMI module included in the saccade BMI (PsychToolbox). During the test sessions, real-time decoder error detection took less than 30 ms. The DED BMI outcome was not cued to the monkeys, only to the saccade BMI. An adaptive LDA algorithm retrained the saccade classifier after a trial was labeled (by the DED BMI) correct(err). In this case, the trial features from the saccade BMI were added to its training dataset (increasing it) and the adaptive LDA decoder was retrained before the next trial. Testing of the DED BMI was performed during 14 sessions for monkey C, and 10 sessions for monkey J.

2.5.3 Implant laterality effect on decoder error detection. After collecting data from the DED BMI, its results were used to asses if the laterality of the implant location had an effect on the classification(err) capabilities of our decoder error detection BMI. Specifically, we wanted to evaluate if it was biased to better classify decoder errors that were made when the target was contralateral to the implant location (as was the case for the saccade BMI). The difference in the proportion of trials correctly and incorrectly classified(err) by the DED BMI, given their laterality, was explored.

First, the laterality of the decoded trials was defined (based on their true target location). Targets 1, 2 and 6 were located in the right visual hemifield; and targets 3, 4, and 5 in the left visual hemifield (see Figure II.2). For monkey C, the ipsilateral and contralateral targets were those located in the left (3, 4, 5) and right (1, 2, 6) visual hemifields, respectively. For monkey J, the ipsilateral and contralateral targets were those located in the right (1, 2, 6), and left (3, 4, 5) visual hemifields, respectively. For more information on the implant location see Figure II.1.

Second, the trials that were correctly(err) and incorrectly(err) decoded were regrouped into two macro subsets. Each one of these macro subsets was further split into ipsilateral and contralateral micro subsets. Finally, the ratio of contralateral (and ipsilateral) trials to the total of trials in each macro subset was computed (of correctly(err) or incorrectly(err) decoded trials). This ratio represents the proportion of contralateral (and ipsilateral) trials in each macro subset. To evaluate if the proportion ratio for each macro subset is significantly different from 0.5 (the case in which the decoder error detection is biased by the laterality of the implant) we computed 10,000 ratios for each macro subset (using bootstrap and resampling from the pool of both ipsilateral and contralateral trials from each macro subset). With these 10,000 ratios we built a probability density function and evaluated if our original contralateral and ipsilateral ratios (per macro subset) are significantly different t the constructed PDF using a p-val of 0.05.

3. Results

This section explores the LFP signals obtained during the saccade BMI trials and their use for real-time decoder error detection. The LFP activity was considered under the two saccade BMI outcomes, the different target locations, and the distance of the incorrectly decoded targets to the true target location. The use of these signals for reliably detecting decoder errors both offline and in real-time, and the effect of the implant laterality on the DED BMI performance were also evaluated. Unless specified, all analysis was done aggregating 11 and 9 sessions from monkey C and J, respectively.

3.1 Temporal, spectral and time-frequency analysis of eLFP

The first step to develop a real-time DED BMI is to explore the characteristic signatures of the eLFP. In this section the error-related LFP signals obtained during the saccade BMI are analyzed in the time and frequency domain. The analysis focuses on the decoder outcome, true target location, and distance to true target. In Table II.1 are presented the number of trials used in the population analysis for both monkeys.

Table II.1. Number of trials for both monkeys for the population analysis.

Here are listed the number of trials used for the population analysis for both monkeys, for both decoder outcome conditions. The list includes the total number of trials, the distance to target (dist1, dist2, dist3), and the number of trials when the previous trial outcome was correct, and incorrect.

Condition Subject	**No. of Correct trials**	**No. of Incorrect trials**	**No. of Correct trials**	**No. of Incorrect trials**
	Monkey C		**Monkey J**	
Total trials	6444	2641	5687	4376
Previous trial correct	4127	1534	3359	2238
Previous trial incorrect	2317	1107	2328	2123
Dist1	-	2372	-	3109
Dist2	-	221	-	1011
Dist3	-	48	-	256

3.1.1 Error-related LFP show decoder outcome specific traits. Initially, the correct and incorrect epochs for all the trials and true targets were compared (using data from an example session for each monkey). From a representative session from monkey C, Figure

II.6 presents the average of the correct (green traces) and incorrect (red traces) epochs, and their error difference (black trace), for all channels in the SEF array. Error bars represent the epochs' S.D. A similar trace across almost all of the SEF channels is found, with some minor amplitude differences. Upon closer inspection of the error difference trace (ErrDiff), some specific traits can be identified. A sharp negative trough around 100ms, a positive peak between 100-300ms, and a wider negative deflection between 250-500ms. These signatures are specific for the SEF array. In the PFC array a minor positive peak between 50 and 100ms is present, as well as a wide negative deflection between 300 and 500ms. In FEF no consistent waveform shape is found, except for a very small negative peak between 200 and 400ms. In monkey C, all channels in a specific array have traces with a very similar shape. For the PFC and FEF arrays the same plots are located in the Appendix.

For monkey J, the average correct and incorrect traces, their standard deviation, and ErrDiff for all the channels in the PFC array are presented in Figure II.7. The same color representation as in Figure II.6 is used. Focusing on the ErrDiff traces, in this array the activity seems to change as we move from top to bottom and left to right. The top left electrodes present some small ripples (0-200ms) with a negative peak around 100ms, followed by a high amplitude peak (200-300) and a small negative trough (300-500ms). In the first two rows of electrodes (and half of the third row, channels 65-77) the ripples are present in all channels while their amplitude and number decrease as the electrodes move from left to right. As the amplitude goes down, the ripples come together to create a negative peak in the same time window (125ms). Likewise, the amplitude of the high peak rapidly decreases from top to bottom and from left to right. One channel stands (electrode 86, middle row on the far right of the array) since its ErrDiff trace has a long lasting negative peak (100-450 ms). This transition in the amplitude of the ErrDiff peaks is present in both SEF and FEF arrays, with the electrode amplitude decreasing from top to bottom (see Appendix).

To further evaluate the similarity of the signals across channels in the same array, the average of all correct and incorrect epochs and channels per array was computed, and its ErrDiff traces were included (Figure II.8). For monkey C, each array has characteristic average traces. In contrast, for monkey J all three arrays seem to have similar mean and ErrDiff traces, closer to the traces found in SEF in monkey C.

A two sample T-test was run to account for the specific differences between correct and incorrect trials, ($p < 0.01$), at each time sample and channel (Figure II.9). This evaluated if the LFP present significant information about the decoder outcome. For monkey C, most of the significant differences lie in the SEF channels between 100-200ms and 300-400ms. For a few PFC channels, this is true in the range 0-100 and 300-450ms. In monkey J, all arrays show electrodes with significant decoder outcome information in the range of 150-300ms and 300-500ms. This is true in the top-left channels in PFC, the bottom channels in SEF, and in almost all FEF channels.

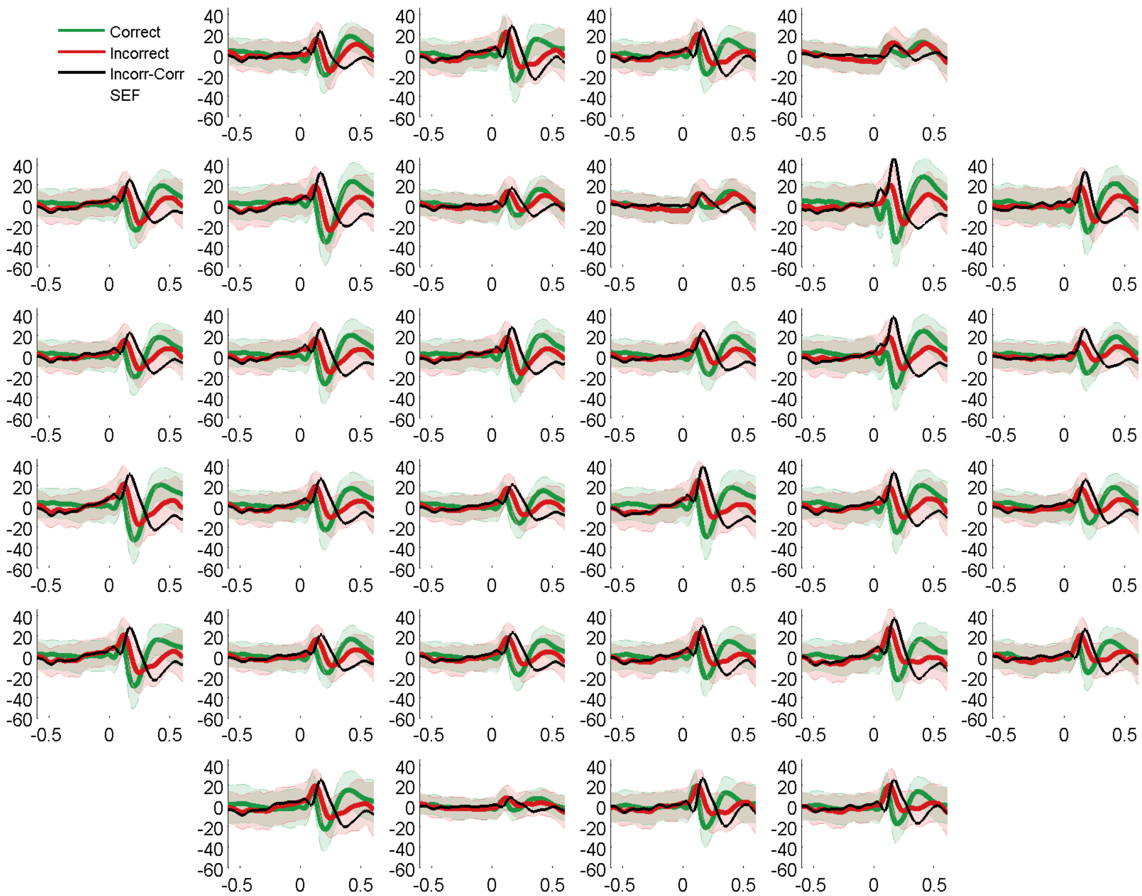


Figure II.6. Average time traces and S.D. for all channels in the SEF array of an example session (monkey C).

A representative session from monkey C demonstrates the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces plotted for the 32 channels in the SEF array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

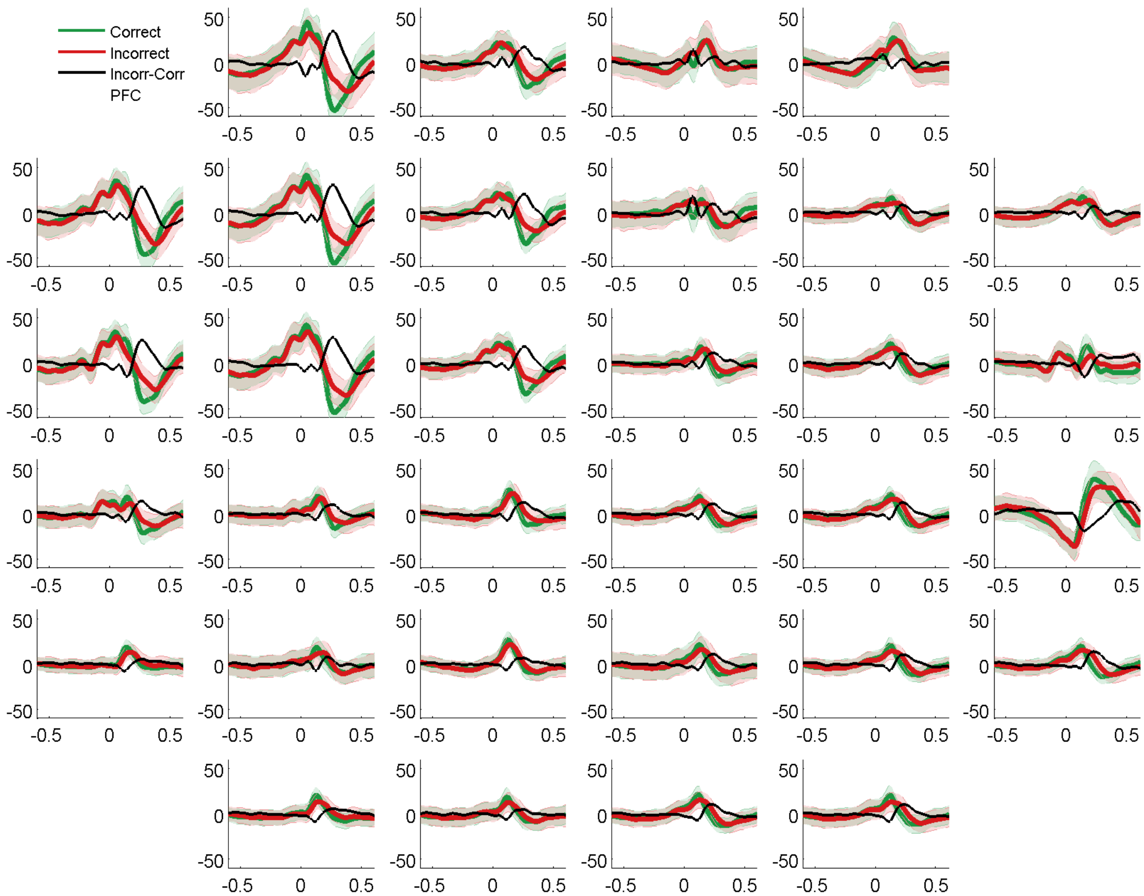


Figure II.7. Average time traces and S.D. for all channels in the PFC array of an example session (monkey J).

A representative session from monkey J demonstrates the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces plotted for the 32 channels in the PFC array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (µV), the X axis time (seconds with respect to feedback onset time).

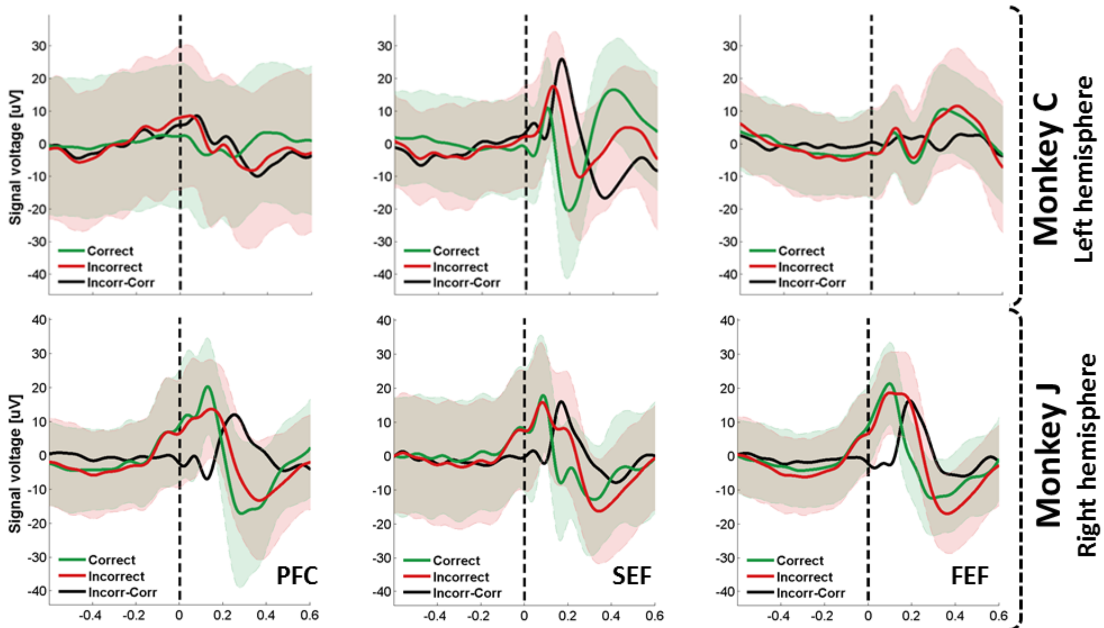


Figure II.8. Average trace (and S.D. error bars) across all epochs and channels in each array of an example session (monkey C and J).

The correct (green trace), incorrect (red trace), ErrDiff (black trace), mean traces for each array are plotted (averages and S.D. taken from all epochs and channels of an example session). The epochs' S.D. is presented as error bars. Y axis represents the voltage (µV), the X axis time (seconds with respect to feedback onset time). The data belongs to a representative session from monkey C (top) and J (bottom). From left to right the arrays are PFC, SEF, and FEF.

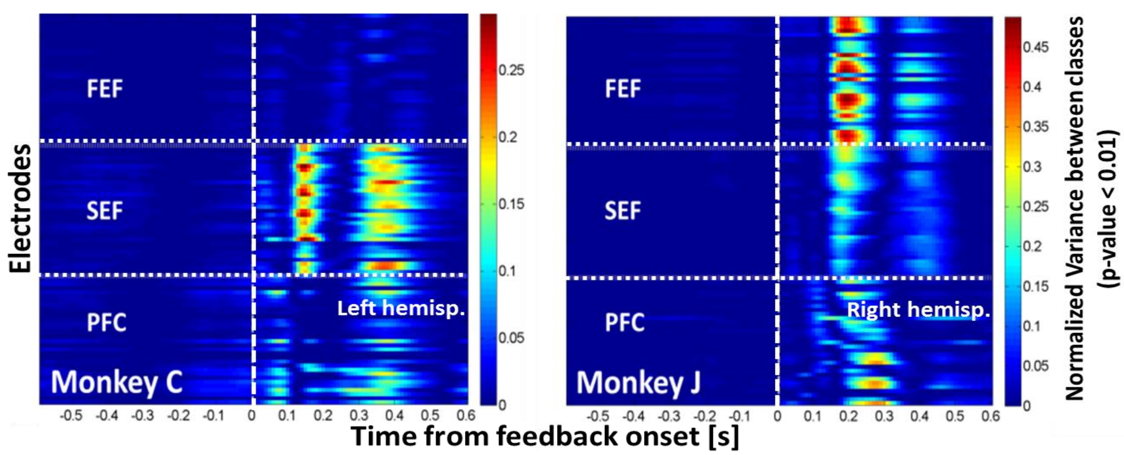


Figure II.9. Significant differences between correct and incorrect trials (monkey C and J).

LFP significance (non-dark blue elements) due to the decoder outcome for a population of monkey C (left) and J (right) sessions. Electrode (channel) numbers are in the Y axis, and time samples in the X axis (time to feedback onset, in seconds). Electrodes 1-32 are for PFC, 33-64 for SEF, and 65-96 for FEF. In monkey C, SEF electrodes show significant information regarding the decoder outcome (two way T-test p << 0.01) between 100-200ms and 300-400ms of most of the channels. Some channels in PFC also represent significant information regarding decoder outcome (0-100, and 300-450ms). In monkey J all arrays show electrodes, in the range 150-300ms and 300-500ms, with significant decoder outcome information. This is true in some PFC and SEF electrode, and in almost all from FEF (two way T-test $p < 0.01$). Colorbar represents the across-trial variance between correct and incorrect trials.

The decoder outcome information could be specific for only some of the true target locations decoded by the saccade BMI (due to the implant laterality). This effect was explored by separating the correct and incorrect trials (from several sessions) by its true target location and comparing their mean and ErrDiff signal. In Figure II.10 and Figure II.11 these traces are presented for all the electrodes in SEF for monkey C and J, respectively. For monkey C, similar average and ErrDiff traces are found for all target locations, with some differences in the peak amplitude, especially larger values for targets 1, 2, 3, and 6. These targets are contralateral to the implant (1, 2, and 6). For monkey J, the average and the ErrDiff signals have a similar shape, and the target location does not relate to big amplitude differences in the signals.

The target-specific differences in the signals are hard to evaluate using only averages and ErrDiff traces. For each target location, a two-sample T-test on all time samples and electrodes was run to evaluate if the differences were significant ($p < 0.01$). The variance between the correct and incorrect trials was plotted in the time samples and channels that were significantly different for both conditions. In monkey C, (Figure II.12) two time windows have significant activity: at 100-200ms and 250-450ms post-feedback onset. The activity in PFC is significant in a few channels for the first time window (in target 3), and in a cluster of electrodes for the second time window (for targets 2, 5, and 6). Overall, contralateral targets show more activity (in both time windows) than ipsilateral targets (see a normalized version of this figure in the Appendix). In monkey J

(Figure II.13), two time window bands with significant activity were present, one at 100-300ms (with peak values between 100-200ms) and another at 300-500ms. The 100-300 time window is very clear in almost all FEF electrodes for all target locations. In SEF, it is present in almost all channels for targets 4, 5, and 6; while only in the bottom ones for targets 1, 2, and 3. The second time window is present in most SEF channels, for target 5, and in some FEF channels for targets 2, 3 and 5. There is no clear laterality effect in this plot (for monkey J).

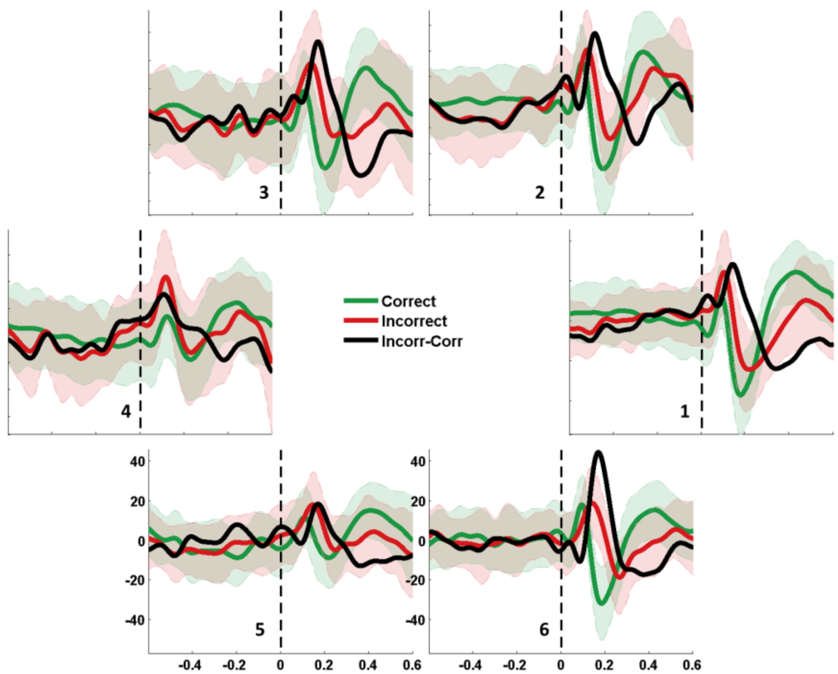


Figure II.10. SEF array mean time traces and S.D. for each target location (monkey C).

Average time traces and S.D. taken from all epochs and channels from a representative monkey C session for each target location. The correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the SEF array are shown. The epochs' S.D. is presented as error bars. The Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

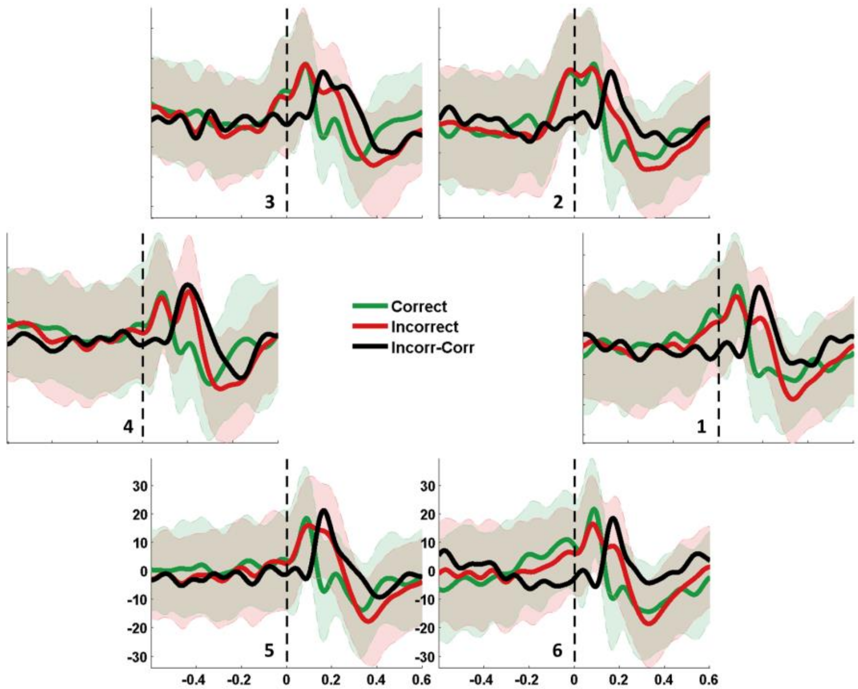


Figure II.11. SEF array mean time traces and S.D. for each target location (monkey J).

Average time traces and S.D. taken from all epochs and channels from a representative monkey J session for each target location. Correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the SEF array are shown. The epochs' S.D. is presented as error bars. The Y axis represents the voltage (µV), the X axis time (seconds with respect to feedback onset time).

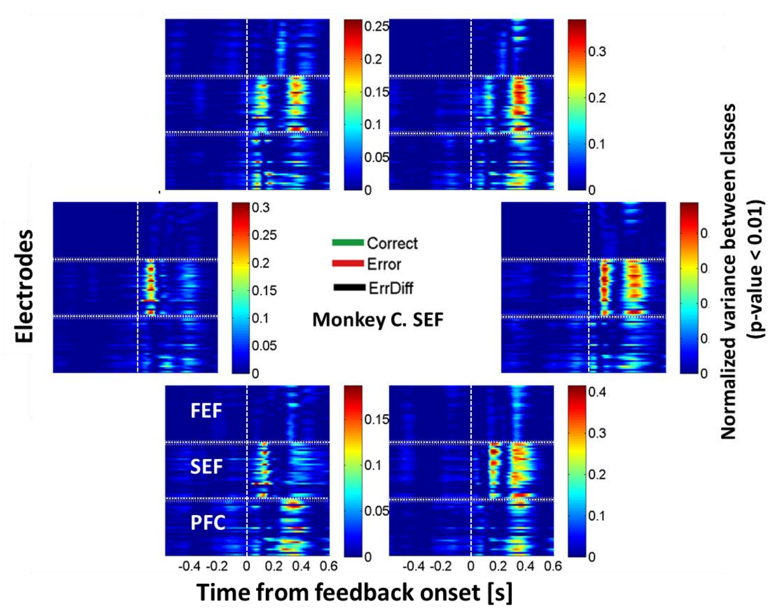


Figure II.12. LFP significant variance for decoder outcome for each target location and array (monkey C).

LFP variance in time samples and channels that is significantly informative of the decoder outcome for monkey C (two sample T-test $p < 0.01$). The Y axis represents electrode (channel) number and the X axis the time to feedback onset. Electrodes 1-32 are for PFC, 33-64 for SEF, and 65-96 for FEF. For all target locations, SEF electrodes, in the range 100-200ms, shows significant decoder outcome information. For targets 1, 2, 3, 6 (with targets 1, 2, and 6 being contralateral to the implants) the same is true in SEF in the range 250-450ms. Targets 1, 2, 5, and 6 present significant information (250-450ms) at some PFC electrodes. The colorbar represents the variance across correct and incorrect trials.

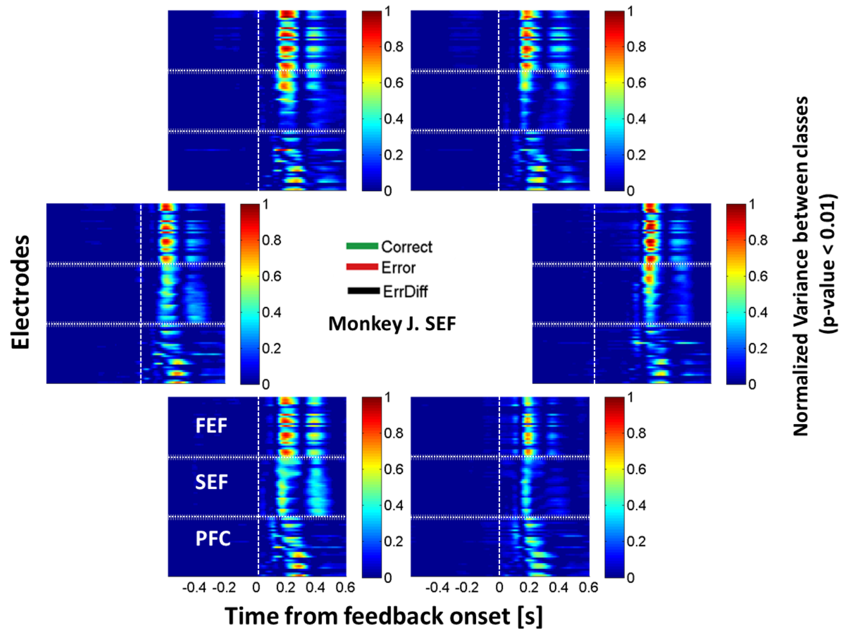


Figure II.13. LFP significant variance for decoder outcome for each target location and array (monkey J).

Monkey J. LFP time samples and channels significantly informative of decoder outcome (two sample T-test $p < 0.01$). The Y axis represents electrode (channel) number and the X axis the time to feedback onset. Electrodes 1-32 are for PFC, 33-64 for SEF, and 65-96 for FEF. The range of variance values are presented in the colorbar. For all target locations most FEF electrodes show, in the range 150-300ms, significant decoder outcome information. In the top-left electrodes from PFC this is also true, but the time window goes from 150-250ms. In the SEF channels, we see a target location dependency: most channels show significant differences for targets 4, 5, and 6 whereas for the remaining target locations this applies only to the bottom-right channels. Between time 300-500ms there is significant activity is found in almost all SEF channels for target 5, while in some electrodes in SEF and FEF, for targets 2 and 3.

3.1.2 Error-related LFP peak amplitude is modulated by the distance to true target.

An important property to evaluate is how the amplitude of the eLFP was modulated by the level of discrepancy (the distance to the true target). This effect was explored by plotting in each target location the mean distance to true target traces (from population data from each monkey) in Figure II.14and Figure II.15. Error bars, representing the population standard deviation, are included.

Since the decoder outcome data depend on the decoder's accuracy, one important element to keep in mind is that the number of correct trials was usually larger than the number of incorrect ones. Moreover, the number of dist1 trials was larger than the number of dist2 trials, and this number was larger than the total of dist3 trials. This is because the saccade BMI classifier tended to make errors closer to the true target, not far away from it (as in the case of dist3 condition).

All the distance to true target plots have the following color code: dark green, light green, orange, and red traces represent the average of correct, dist1, dist2, and dist3 epochs, respectively. In Figure II.14 the average traces from the SEF channels are shown (for a population of monkey C sessions). The positive peak amplitude of dist3 seems to be larger than the amplitude of dist2 and dist1 (targets 3, 4, 5 and 6). The negative deflection also appears to have larger amplitude for targets 3, 4 and 5 (ipsilateral to the implant). For all targets except 4 and 6, a considerable number of dist3 traces are present. For monkey J, the distance to true target traces are more similar in shape and amplitude. In Figure II.15 are plotted the average traces, from a population of monkey J sessions, from an SEF channel. The modulation of amplitude due to distance to true target can be seen in targets 1, 2, and 6, for both the positive and negative deflections, but mainly in target locations 2 and 6 (ipsilateral to the implant).

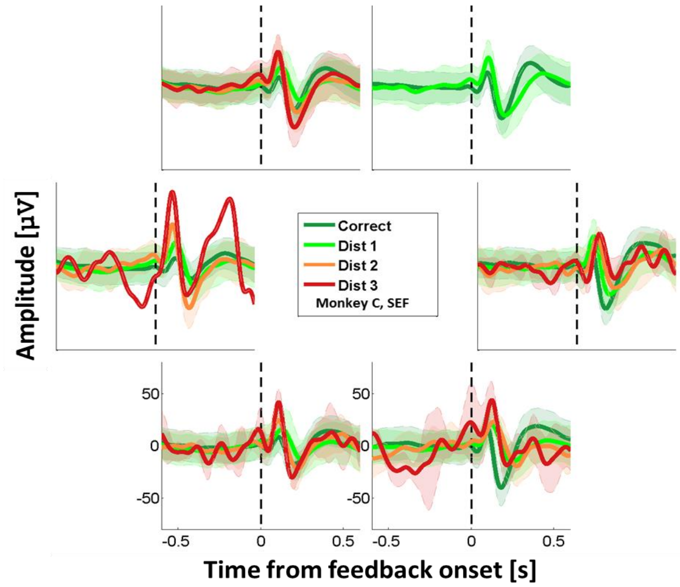


Figure II.14. Population distance to true target for each target location in 1 SEF channel (monkey C).

Average distance to true target (across all epochs from a population of monkey C sessions) for each target location at one SEF channel. Dark green, light green, orange, and red traces represent the correct, dist1, dist2, and dist3 mean traces, respectively. For most of the targets the positive peak amplitude seems to be modulated by the distance to true target, with a larger amplitude for larger distance to true target. Error bars represent the population standard deviation.

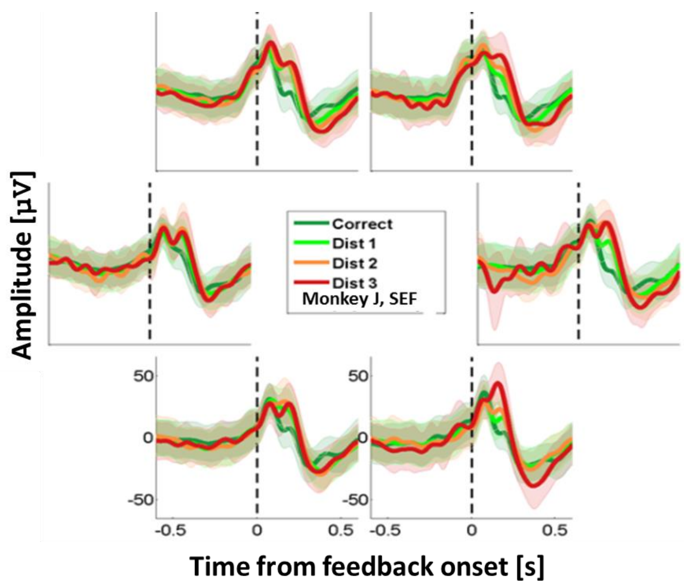


Figure II.15. Population distance to true target for each target location in 1 SEF channel (monkey J).

Average distance to true target (across all epochs from a population of monkey J sessions) for each target location at one SEF channel. Dark green, light green, orange, and red traces represent the correct, dist1, dist2, and dist3 mean traces, respectively. For most of the ipsilateral targets (1, 2, 6) the positive peak amplitude seems to be modulated by the distance to true target, with a larger amplitude for larger distance to true target. Error bars represent the population standard deviation.

For each time sample and channel, from each monkey's population data, a one-way ANOVA was run across all distance to target conditions (Figure II.16). For monkey C, all SEF channels in the 150-250ms window have significant information about the distance to true target ($p < 0.01$, Bonferroni corrected). On the other hand, in monkey J, the channels at the bottom of the SEF array show significant information (and high explained variance) for the distance to true target in the range 100-250ms and 350-450ms ($p < 0.01$, no Bonferroni correction was applied). In FEF some electrodes present significant activity between 220 and 280ms, and 350 and 450ms. In the PFC array some channels present it as well in the range 350 and 450ms. Some electrodes in PFC show significant activity in the range 170-250ms.

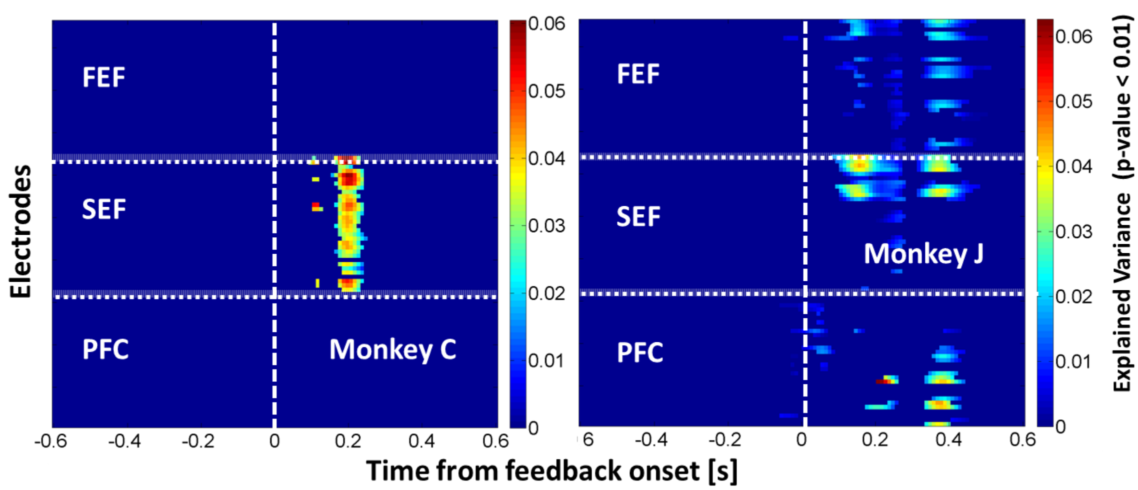


Figure II.16. One-way ANOVA for distance to true target for dist1, dist2, dist3 (monkey C and J).

LFP explained variance for distance to true target for a population of monkey C (left) and J (right) sessions. Y axis represents electrode (channel) number and X axis the time to feedback onset. Electrodes 1-32 are for PFC, 33-64 for SEF, and 65-96 for FEF. In monkey C SEF channels show significant differences for the distance to true target in the range 170-250ms ($p < 0.01$, no Bonferroni correction was applied). In monkey J the bottom-right channels in the SEF array show significant differences in the range 100-250ms, and most of FEF channels show this property in the range 100-250 and 350-450ms post-feedback onset ($p < 0.01$, no Bonferroni correction was applied). Explained variance values are presented in the color bar.

3.1.3 The prior trial outcome influences the amplitude of the eLFP signal. Target location and distance to true target are not the only conditions that can influence the eLFP. It has been shown that a previous incorrect trial affects the following trial. Bootstrap (1000 iterations) analysis of a two sample T-test ($p < 0.01$, Bonferroni corrected) was performed to check the effect of the previous trial outcome condition (when the previous trial, $T = n-1$, was correct or incorrect) on the current trial (at trial $T = n$).

Significant differences due to the trial outcome of the previous trial (correct or incorrect at $T = n-1$) were found when the current trial ($T = n$) was incorrect (Figure II.17, top row). Significant differences were found between 450 and 550ms post-feedback onset in some SEF electrodes in monkey C and J. When the effect of previous correct and incorrect trials (at $T = n-1$) is evaluated on correct trials ($T = n$), significant differences (Figure II.17, bottom row) were found between 200 and 350ms and 450 and 550ms post-feedback onset in some SEF electrodes of monkey C. In monkey J, on the other hand, this effect was found between -50 and 0ms post-feedback onset in several SEF electrodes (mainly in the top right of the array), and between 100 and 250ms in the top right electrodes of the PFC and SEF arrays, and in several FEF electrodes.

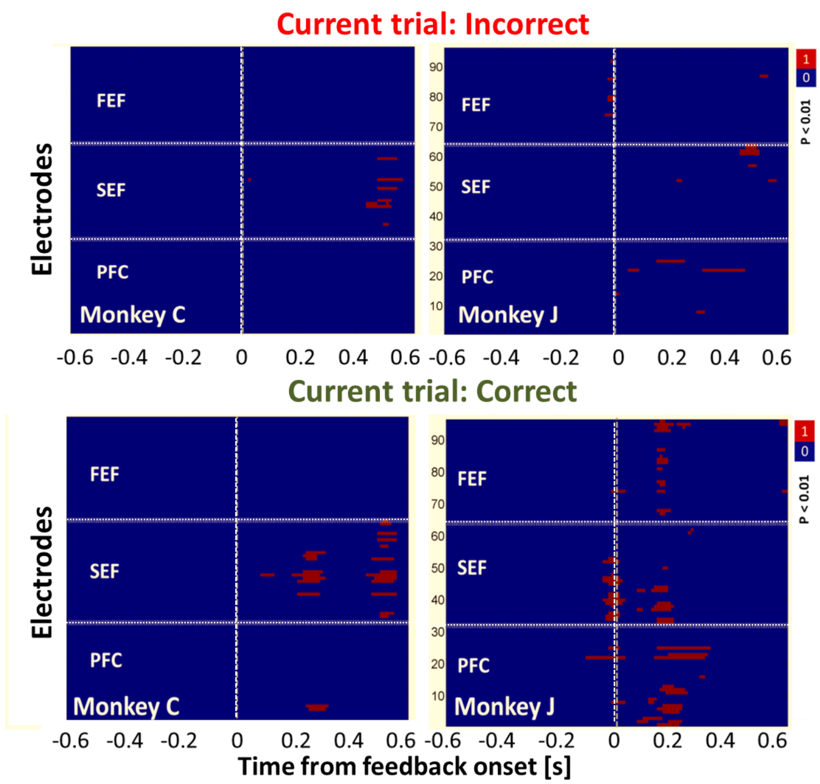


Figure II.17. Effect of previous trial outcome on the eLFP

Average result from 1000 iterations bootstrap analysis of a two sample T-test ($p < 0.01$, Bonferroni corrected). The test compared the current trial outcome (at trial t = n, incorrect in the top row, correct in the bottom row) when the previous trial (t = n-1) was either correct or incorrect. Significant differences (in red) were found after the feedback onset of the current trial (t = n). An effect of previous trial outcome on the recorded eLFP (top row) was found between 450 and 550ms post-feedback onset in some SEF electrodes of monkey C and in two bottom right electrodes of monkey J ($p < 0.01$). The effect of previous trial outcome on the recorded correct trials (bottom row) was found between 200 and 350ms and 450 and 550ms post-feedback onset in some SEF electrodes of monkey C. In monkey J this effect was found between -50 and 0ms post-feedback onset in the top right SEF electrodes, and between 100 and 250ms in some electrodes of PFC, SEF and FEF.

3.1.4 Error-related LFP contain information about decoder outcome at different frequency bands.

The log power of the difference between the average incorrect and correct spectrograms was explored to find possible activity related to decoder outcome (average across 1000 bootstrap iterations). Figure II.18 presents the power of the difference (ErrDiff) of the average incorrect and correct spectrograms for a population of monkey C sessions (11 sessions). All channels, in all arrays, show a strong decrease in power in the 15-30Hz band from 200ms to 600ms post-feedback presentation (and this effect continues until the end of the epoch). Above 50Hz, an increase in power is present in all arrays from 300-700ms post-feedback onset. Also, in PFC and FEF all channels and arrays show a strong increase in power at the 1-10Hz band between 200-600ms (see Appendix). In SEF all channels increase power between 200-400ms, then decrease it for 100ms, and increase it again from 500-700ms post-feedback onset.

In monkey J (Figure II.19), a similar phenomenon occurs. In all channels and arrays we find a strong power decrease between 10-30Hz that starts at 100ms post-feedback onset and that shrinks in the frequency band from 30 to 25Hz at 200ms. A similar decrease occurs in the 30-40Hz between 300-700ms. In the period 150-600ms post-feedback onset an increase of power is found in the 1-10Hz band. In the other two arrays we find similar changes in the frequency power, with a decrease of power in the beta band, and an increase of activity in the alpha band (see appendix).

In the gamma (30-80Hz) and high gamma (80-200Hz) bands interesting features are present in both monkeys. In the 60-200 Hz band (Figure II.20), each frequency bin is scaled [0,1] across all time samples to highlight the features across all frequencies. In monkey C, SEF and FEF channels show a decrease (100-300ms post-feedback onset time) and an increase in power (300-600ms post-feedback onset time). In the PFC array we find a decrease (0-300ms) and then an increase in power (300-600ms). In monkey J, PFC, SEF and FEF channels show a decrease in power (300-500ms post-feedback onset), then immediately an increase in power for 200ms (500-700ms). The PFC channels present also an increase in activity (0-100ms post-feedback onset time).

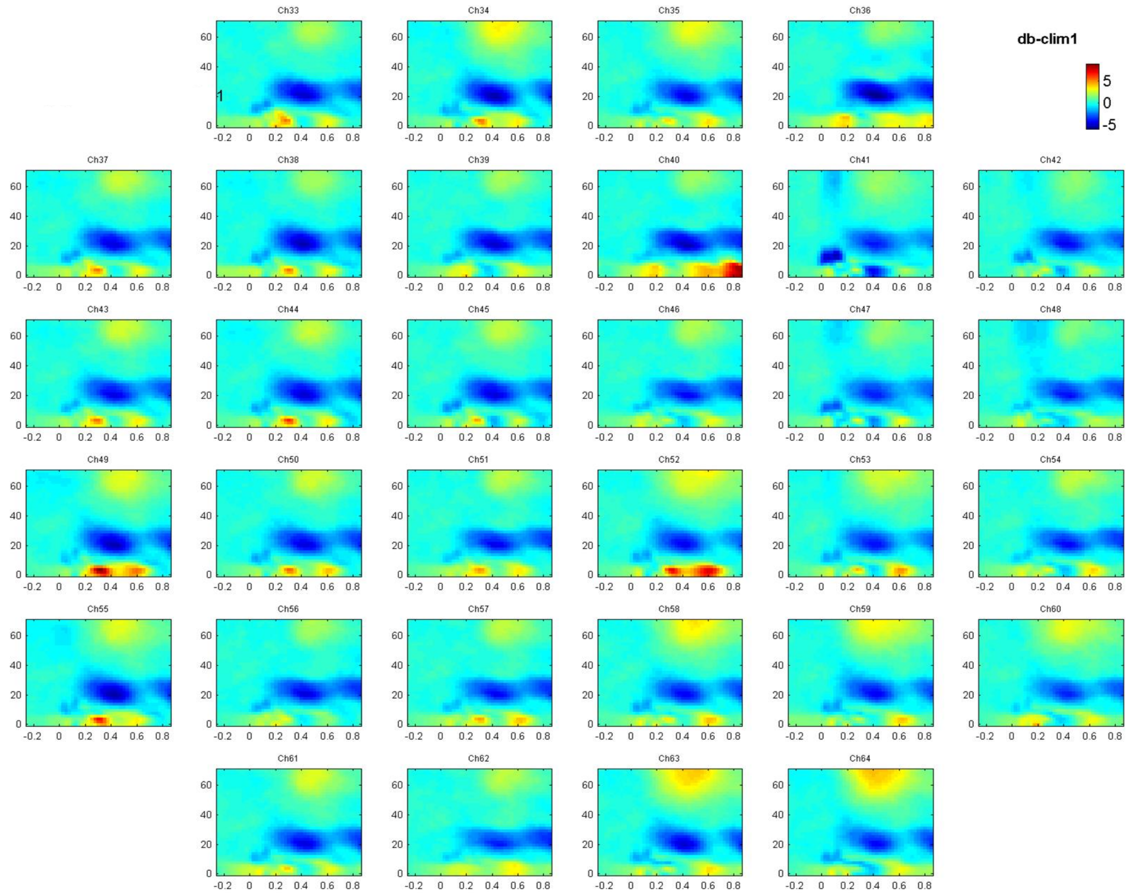


Figure II.18. ErrDiff normalized spectral power in SEF array for monkey C.

Log power of the difference of the incorrect and correct mean spectrogram for a population (11 sessions) of monkey C sessions. All channels belong to SEF. Log power measured in dB and the color bar scale is normalized for all channels. In the 15-30Hz band, activity strongly decreases while increases are found in the 1-10 Hz band between 200-400ms and 500-700ms post-feedback onset. A decrease in activity is also present immediately after the feedback onset, lasting 200ms and covering frequencies between 10-20Hz.

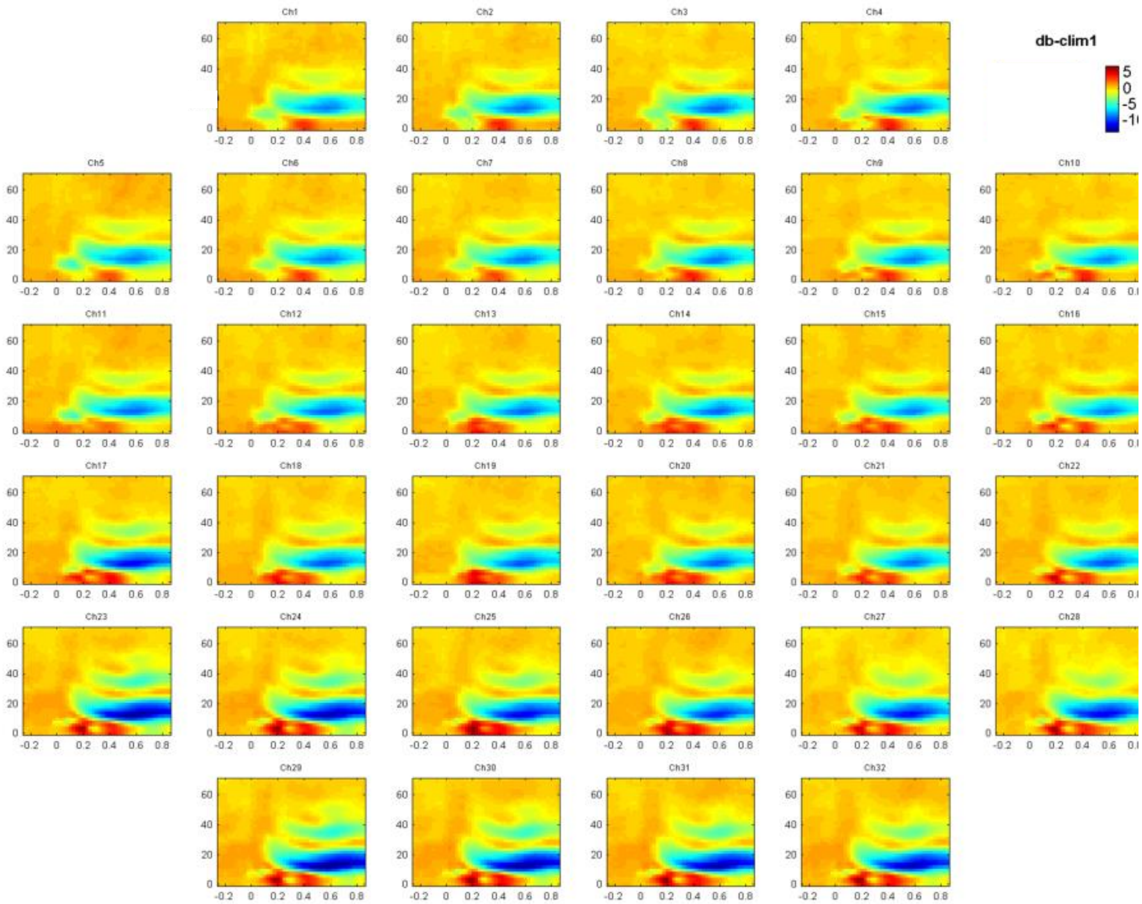


Figure II.19. ErrDiff spectral power in SEF array for monkey J.

Log power of the difference of the average incorrect and correct spectrogram for a population (9 sessions) of monkey J sessions. All channels belong to SEF. Log power measured in dB and the color bar is equal to all the channels. The activity in the 10-25 Hz band strongly decreases between 200-800ms post-feedback onset. In the period 150-600ms post-feedback onset activity in the 1-10Hz band strongly increases.

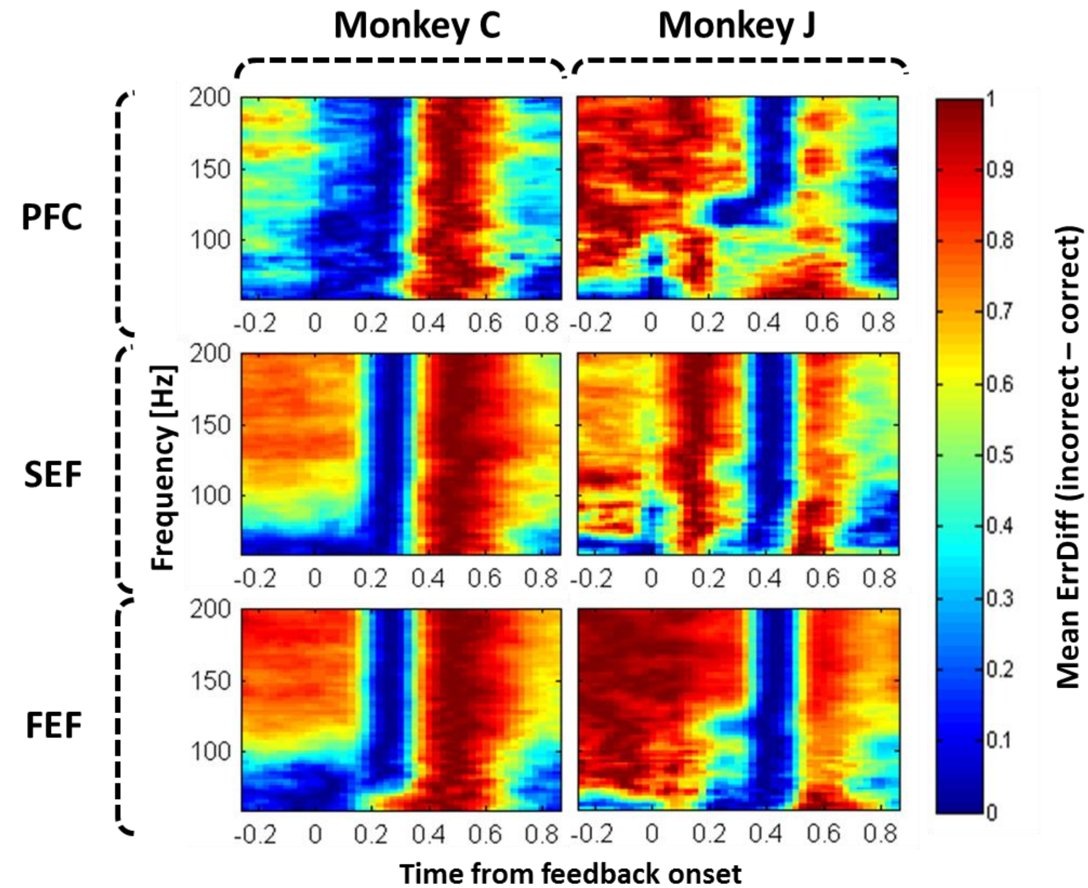


Figure II.20. Mean ErrDiff spectral power (scaled over each freq. bin) over high gamma for one channel in each array.

High gamma spectral power difference between the average (from 1000 bootstrap iterations) incorrect and correct spectrogram from a population of sessions from monkey C (11 sessions, left column) and monkey J (9 sessions, right column). Top, middle and bottom rows represent a single channel from PFC, SEF and FEF, respectively. The X axis shows the time to feedback onset (in seconds), and the Y axis the frequency (in Hz). The color bar represents scaled values [0,1] across time for each frequency bin. Cool (blue) and hot (red) colors represent low and high values, respectively. In **monkey C,** SEF and FEF channels show in the 50-200Hz band, a decrease (100-300ms post-feedback onset time) and an increase in power (300-600ms post-feedback onset time). In PFC is found a previous decrease (0-300ms) followed by an increase in power (300-600ms). In **monkey J**, PFC, SEF and FEF channels show in the 50-200 Hz band, a decrease in power in 300-500ms post-feedback onset, then immediately an increase in power for 200ms (500-700ms). PFC channels, in the 50-200Hz band, present also an increase in activity (0-100ms post-feedback onset time).

To inspect if the power at a set of frequency bands was significantly informative about the decoding outcome, mean power values across the delta, theta, alpha, beta, gamma and high gamma bands were computed for all the correct and incorrect trials of the population sessions (for each monkey). The results of a two sample T-tests for each frequency band is presented in Figure II.21 for each monkey and for each array's top left (odd rows) and bottom right channels (even rows).

Top rows represent the significant (red) frequency bands and time samples (from 1000 bootstrap iterations, p-val < 0.05, Bonferroni corrected). In monkey C, beta and gamma are the most significant frequency bands. In monkey J, all the frequency bands present significant differences; especially alpha, beta and high gamma. The bottom rows represent the variability between correct and incorrect trials for each frequency band and time point (from the same 1000 iterations, plotting only the significant values, p-val < 0.05, Bonferroni corrected). Hot (red) and cool (blue) colors represent large and small differences in decoding outcome classes (correct and incorrect trials), respectively. The variability values are normalized per array and only non-dark blue colors are significant. In monkey C (first three rows) the beta, gamma and high gamma are significant frequency bands after the feedback onset (with beta and high gamma showing the largest differences in variance across correct and incorrect trials). Monkey J (last three rows) presents significant activity with the largest differences between correct and incorrect trials in the alpha and beta bands, and with a short period of activity in the delta and theta band (200ms post-feedback onset). For each monkey and array, all channels show similar behavior.

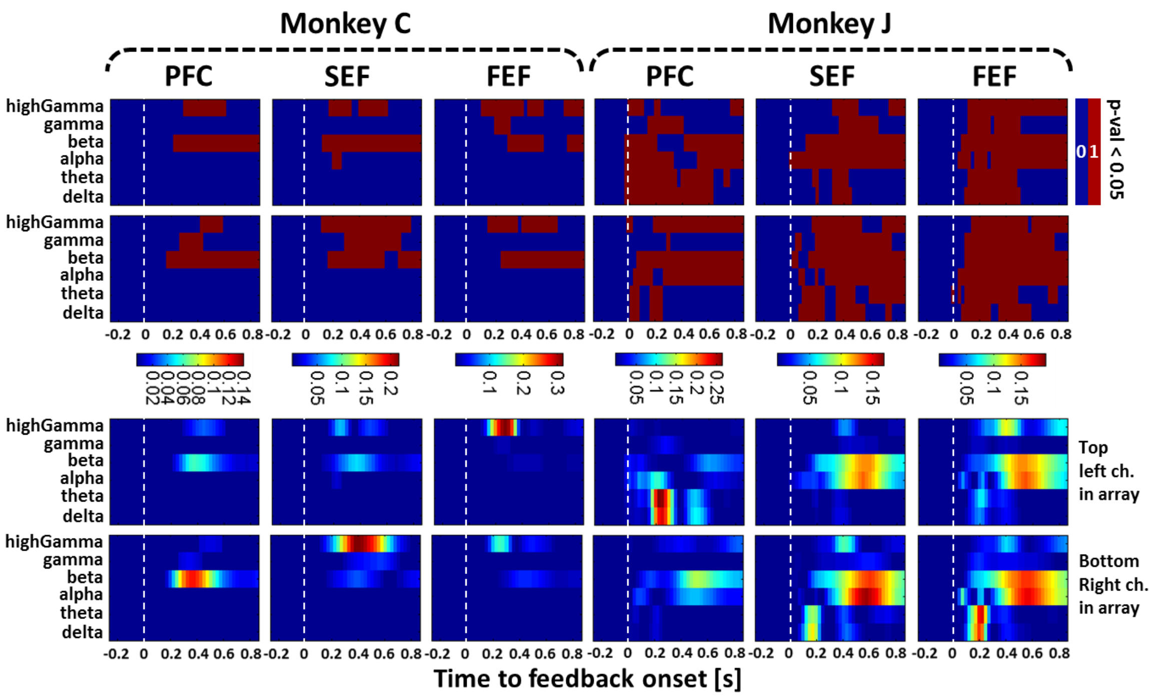


Figure II.21. Frequency bands with significant information about decoding outcome.

Two sample T-test between correct and incorrect trials for all frequency bands (y axis: delta, theta, alpha, beta, gamma, high gamma) of a population of sessions from monkey C (11 sessions, first three rows from left to right), and monkey J (9 sessions, last three rows from left to right). X axis represents the time to feedback onset. Top rows represent the significant (red) frequency bands and time samples. In monkey C, beta and gamma are the most significant frequency bands. In monkey J, the most significant frequency bands are alpha and beta, with a short period for delta and theta. White dashed lines mark the feedback onset time (vertical).The bottom rows represent the variability between correct and incorrect trials for each frequency band and time point. Hot (red) and cool (blue) colors represent large and small differences in decoding outcome classes (correct and incorrect trials), respectively. Variance values are normalized per array and only values significant are presented. Error bars represent percentage of variability between correct and incorrect trials. All significant values obtained after 1000 bootstrap iterations, p-val < 0.05, Bonferroni corrected.

3.1.5 Cross-frequency amplitude-amplitude coupling changes. The results in the previous section suggest that cross-frequency coupling might be happening (between alpha and beta, and beta and high gamma). With data from the 6 frequency bands, the cross-frequency amplitude-amplitude coupling was computed (averaging 1000 bootstrap iterations) for the correct and incorrect trials for the period before and after the feedback onset (from each monkey’s population sessions). The periodDiff (difference between the pre-onset and post-onset periods) cross-frequency coupling of incorrect and correct trials was computed to get the ErrDiff-periodDiff cross-frequency coupling.

For monkey C, (Figure II.22) no strong changes in cross-frequency coupling for any of the frequencies were found in the arrays except for a decrease in delta-alpha and theta-alpha coupling in PFC and SEF. The average (across all the array electrodes) delta-alpha coupling decrease across PFC channels was 0.049 (±0.086 S.D.) whereas in SEF it reached 0.085 (±0.037 S.D.). The average theta-alpha coupling decrease across PFC channels was 0.041 (±0.08 S.D.) whereas in SEF it reached 0.077 (±0.035 S.D.). Only two channels in PFC (18 and 27) showed changes in other coupling frequencies. Channel 18 showed an increase in delta-alpha (0.215), theta-alpha (0.214), delta-gamma (0.167), theta-gamma (0.16), delta-highGamma (0.154) and theta-highGamma (0.144) coupling. Channel 27 presented an increase in delta-highGamma (0.395), delta-gamma (0.396), delta-beta (0.358), theta-highGamma (0.398), theta-gamma (0.409), theta-beta (0.377), alpha-highGamma (0.40), alpha-gamma (0.511), alpha-beta (0.544), and beta-highGamma (0.179) coupling.

In monkey J, (Figure II.23) the changes in coupling were found in all the arrays but mainly in PFC and FEF. We observed an increase in delta-alpha (0.106 ±0.094 S.D. in PFC channels, and 0.141 ±0.046 S.D. in FEF channels) and theta-alpha (0.12 ±0.1 S.D. in PFC channels, and 0.128 ±0.052 S.D. in FEF channels) coupling whilst a decrease in alpha-beta (0.321 ±0.113 S.D. in PFC channels, 0.181 ±0.034 S.D. in SEF channels, and 0.247 ±0.058 S.D. in FEF channels), alpha-gamma (0.221 ±0.119 S.D. in PFC channels, 0.215 ±0.06 S.D. in SEF channels, and 0.272 ±0.124 S.D. in FEF channels) and beta-gamma (0.106 ±0.082 S.D. in PFC channels, 0.11 ±0.03 S.D. in SEF channels, and 0.14 ±0.032 S.D. in FEF channels) coupling.

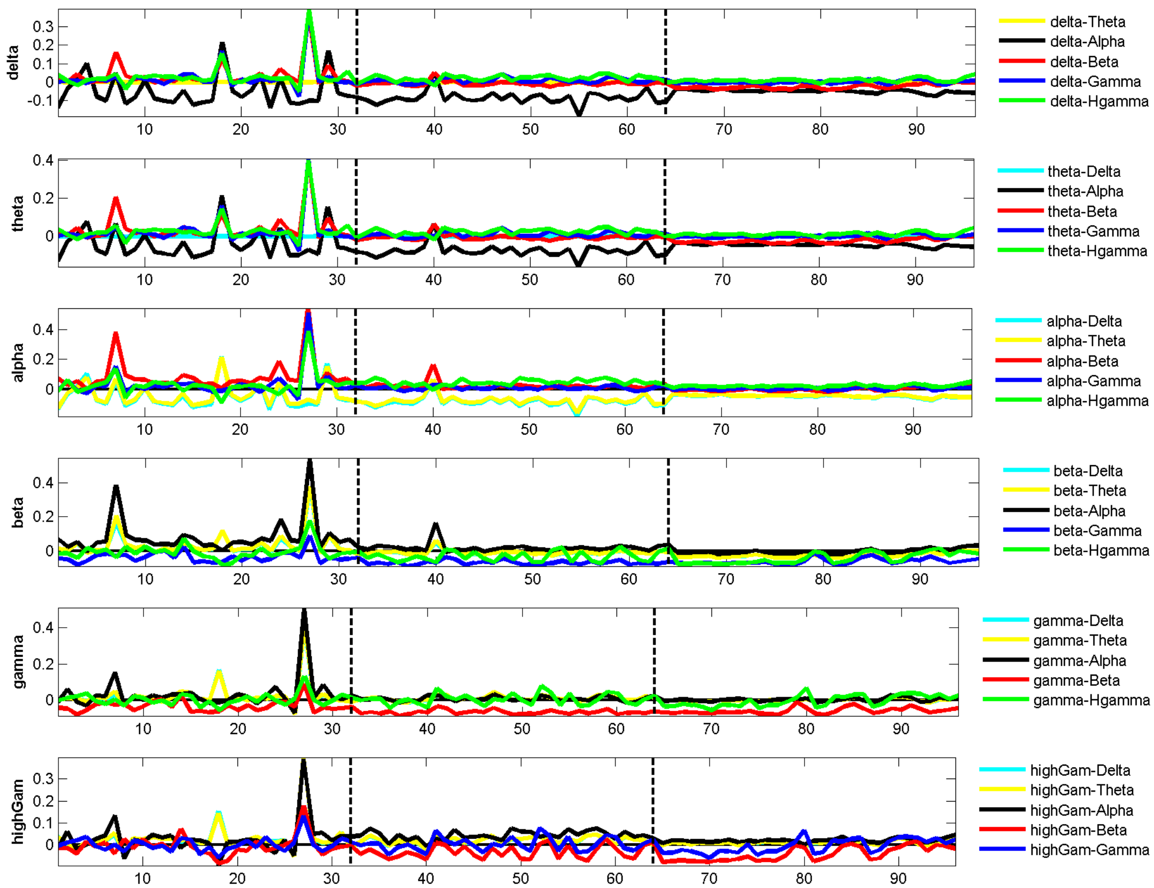


Figure II.22. ErrDiff-periodDiff cross-frequency amplitude-amplitude coupling for all frequency bands (monkey C).

Average difference between the incorrect and correct periodDiff (with periodDiff defined as the difference between the post and pre-feedback period cross-correlation) for all channels and arrays from a population (11 sessions) of monkey C sessions (average across 1000 bootstrap iterations). The X axis represents the channel number. The channel distribution is: PFC 1-32, SEF 33-64, FEF 65-96. The Y axis represents the difference in the cross-correlation coupling values. From top to bottom, the main frequency band is delta, theta, alpha, beta, gamma, and high gamma. Each frequency band is coupled with the delta (cyan trace), theta (yellow trace), alpha (black trace), beta (red trace), gamma (blue trace), and high gamma (green trace) band. Dotted lines represent a change in array. A decrease in delta-alpha and theta-alpha coupling is found in the PFC and SEF arrays.

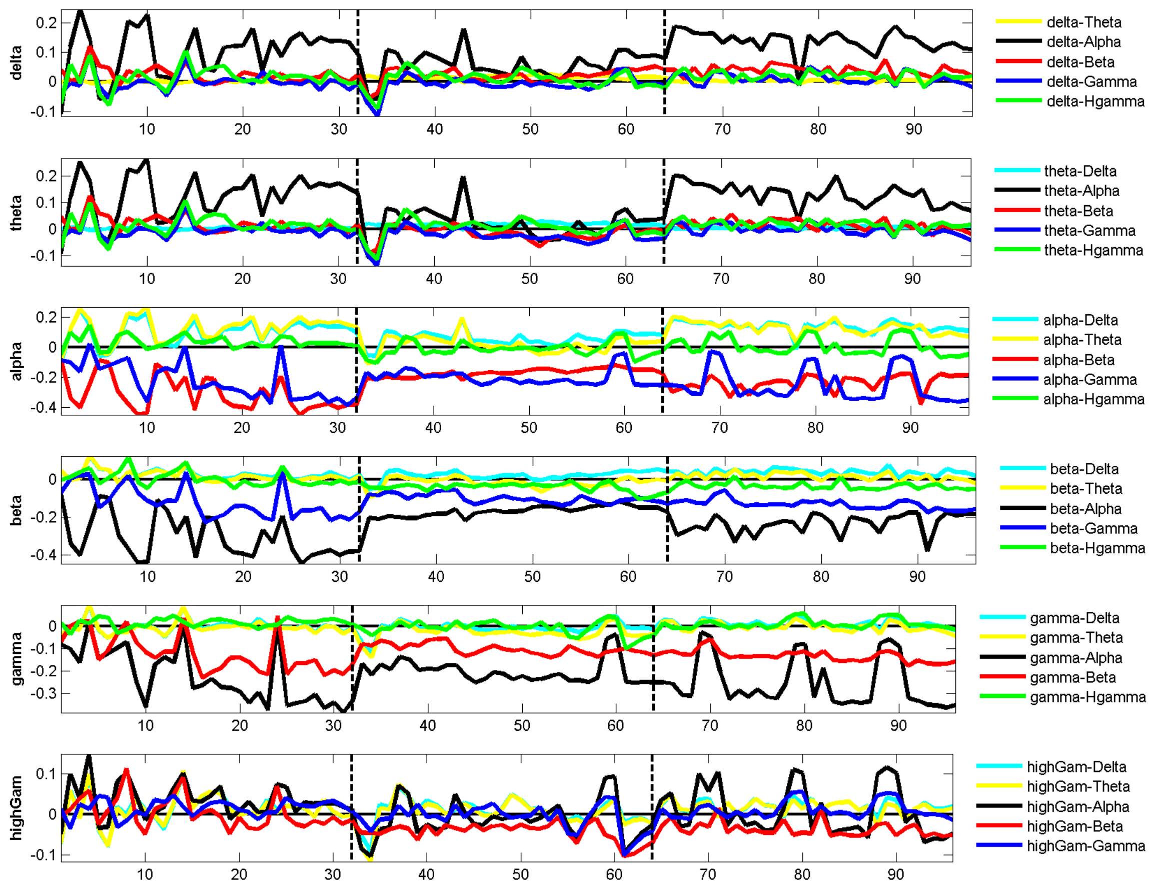


Figure II.23. ErrDiff-periodDiff cross-frequency amplitude-amplitude coupling for all frequency bands (monkey J).

Average difference between the incorrect and correct periodDiff (with periodDiff defined as the difference between the post and pre-feedback period cross-correlation) for all channels and arrays from a population (9 sessions) of monkey J sessions (average across 1000 bootstrap iterations). The X axis represents the channel number. The channel distribution is: PFC 1-32, SEF 33-64, FEF 65-96. The Y axis represents the difference in the cross-correlation coupling values. From top to bottom, the main frequency band is delta, theta, alpha, beta, gamma, and high gamma. Each frequency band is coupled with the delta (cyan trace), theta (yellow trace), alpha (black trace), beta (red trace), gamma (blue trace), and high gamma (green trace) band. Vertical Dotted lines represent a change in array. An increase in delta-alpha and theta-alpha coupling was found as well as a decrease in alpha-beta, alpha-gamma and beta-gamma coupling.

3.2 Offline decoder error detection

In order to determine the best features and parameters we considered two different approaches, one employing 10-fold cross-validation of multiple sessions and one focusing on using different sessions for training and testing, with a similar design to the online DED BMI tests.

3.3.1 Cross-validation analysis: To begin, we ran multiple iterations of all the possible values that our four different parameters could take in 65 monkey C sessions and 9 monkey J sessions. Via 10-fold cross-validation, the true positive (only incorrect trials), true negative (only correct), and overall classification percentage of the DED BMI were evaluated. Table II.2 presents a summary of the 4 different parameters iterated.

Table II.2. Parameters iterated in the offline 10-fold cross-validation analysis.

Here are listed the different parameters iterated during the 10-fold cross-validation analysis for 65 monkey C sessions and 9 monkey J sessions. From these parameters the ones providing the best performance accuracy were selected for the real-time DED BMI sessions.

Parameter	Possible Values					
Baseline removal	0Base	1Base	-	-	-	-
Array selection	SEF only	PFC only	SEF and PFC	PFC and FEF	SEF, PFC and FEF	
Features extraction function	Mean	Mean2	minMax	-	-	-
Feature vector transformation function	None	Log	sqr	sqrt	mean	zscore

For monkey C, the median DED BMI performance values for all parameters are presented in Figure II.24. These results suggest that for monkey C's data we get the highest decoder(err) performance when no baseline is removed and only channels from the SEF array are selected. Also, this analysis shows that we should compute the mean of the data windows to extract the features, and later z-score them. In preliminary analysis using only the FEF, FEF and SEF, and FEF and PFC arrays, the average decoder(err) performance was the lowest. For this reason these array combinations are not shown in the plots. All the feature transformation functions but ‘log’ and ‘sqr’ provided similar decoder(err) performance values. The ‘log’ and ‘sqr’ functions presented a decrease in performance by approximately 20%. In Figure II.25 are the average decoder(err) performance values across 9 monkey J sessions. These results indicate the highest performance was obtained when 1) the baseline is not removed, 2) the FEF and PFC arrays are selected, 3) the mean function is applied to extract the features, and 4) no transformation is applied. As with the results with monkey C, the ‘log’ and ‘sqr’ feature transformation functions had the poorest decoder(err) performance values (by more than 20%).

Upon optimizing the parameters for each monkey (using the 10-fold cross-validation approach), the decoder(err) performance for those parameters using 10-fold cross-validation were plotted. Figure II.26 present the mean performance values for monkey C. The left Y axis has the performance values for the saccade BMI (blue trace). The right Y axis has the scale for the DED BMI decoder(err) performance. The green, red, and black traces correspond to the true negative (no decoder error), true positive (decoder error), and overall decoder(err) performance, respectively. In general, the decoder is very stable across sessions. We found 5 sessions with performance values below 90% between sessions 33 and 46 (between the dates 121114 and 130415, YYMMDD), and only 3 of those were below 80%. The overall good saccade decoder performance (68.3%) provided, for the decoder(err) training, more correct trials than incorrect ones. This biased our DED decoder(err), an effect that can be seen in improved performance for detecting true negatives (correct trials). Overall, the decoder(err) performance across all monkey C sessions was 94.8% ± 4 S.D. If we remove the low performance outliers (between sessions 33 and 46) the performance increases to 96%.

Likewise, in Figure II.27 are shown the mean and median (from the 10-fold CV) decoder(err) performance values for monkey J sessions. Continuous and dash lines represent the mean and median, respectively. The blue Y axis and the blue trace correspond to the saccade BMI decoder performance. The DED BMI decoder(err) performance values correspond to the right Y axis. The green, red, and black traces identify the true negative (no decoder error), true positive (decoder error), and overall DED BMI decoder(err) performance, respectively. For monkey J, the performance of the saccade BMI was close to 50%, giving a similar number of correct and incorrect trials. Nevertheless, the true negative results still presented slightly better decoder(err) performance. Overall, the decoder(err) performance across all monkey J sessions was 94.9% ± 1.4 S.D

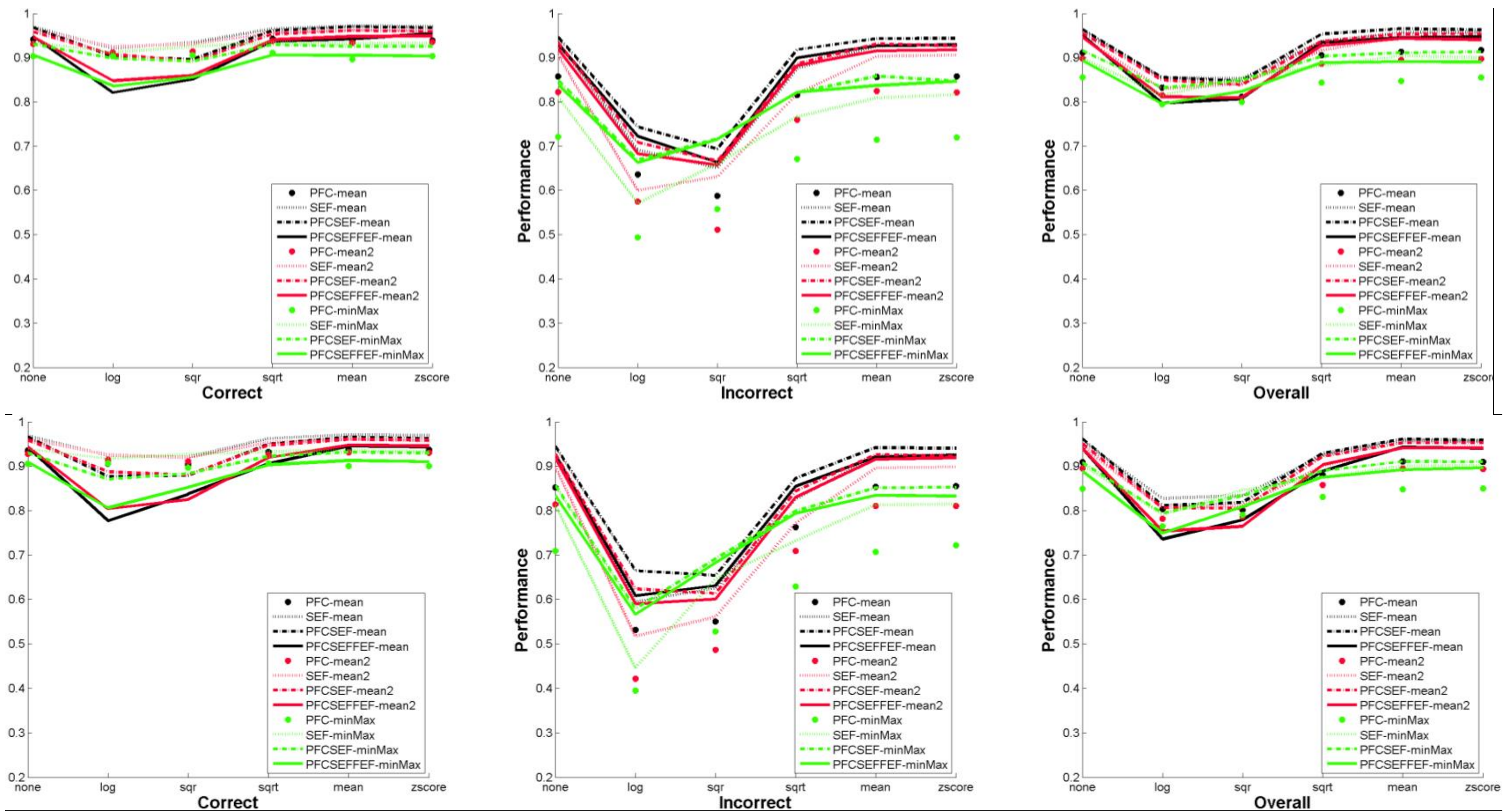


Figure II.24. Analysis of decoding(err) parameters via 10-fold cross-validation for monkey C.

Offline DED BMI median classification performance across 65 monkey C sessions under different parameters. The results are presented for correct (left column), incorrect (middle column), and overall trials (right column). Baseline removal was applied only on bottom plots. Black, red, and green colors represent values for the feature extraction functions: mean, mean2, and minMax; respectively. Line styles represent values for the array combination. Dots, light lines, dash lines, and dark lines represent PFC, SEF, PFC –SEF, and PFC-SEF-FEF arrays, respectively. The different feature extraction functions (none, log, sqr, sqrt, mean, and zscore) are represented in the x axis.

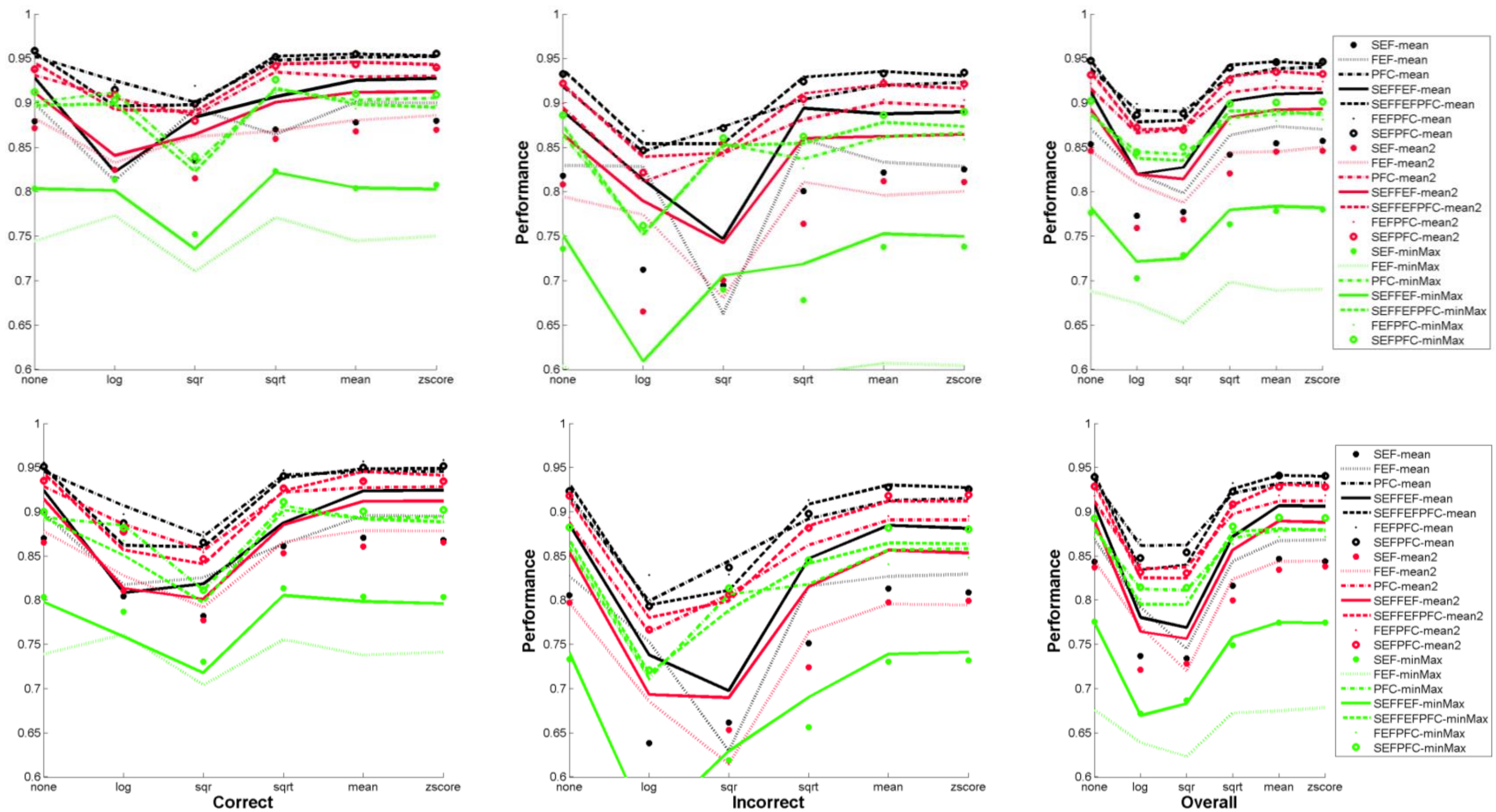


Figure II.25. Analysis of decoding(err) parameters via 10-fold cross-validation for monkey J.

Offline DED BMI mean classification performance across 9 monkey J sessions under different parameters. The results are presented for correct (left column), incorrect (middle column), and overall trials (right column). Baseline removal was applied only on the bottom plots. Black, red, and green colors represent values for the feature extraction functions: mean, mean2, and minMax; respectively. Line styles represent values for the array combination. Big dots, light lines, dash lines, dark lines, dash and dot lines, small dots, and circles represent SEF, FEF, PFC, SEF-FEF, SEF-FEF-PFC, FEF-PFC, SEF-PFC arrays, respectively. The different feature extraction functions (none, log, sqr, sqrt, mean, and z-score) are represented in the x axis.

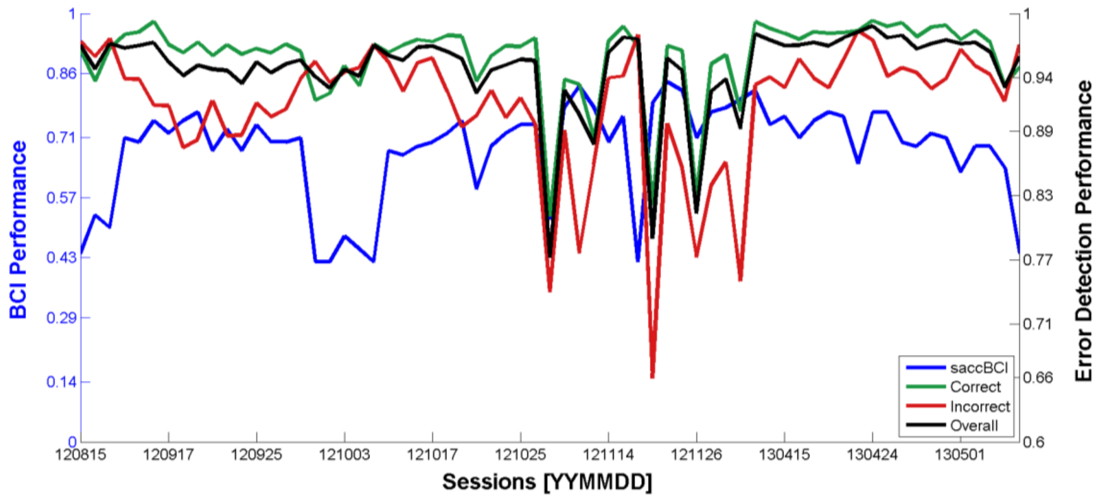


Figure II.26. Decoder(err) performance using 10-fold CV and optimized parameters (monkey C).

10-fold CV decoder(err) performance across 65 monkey C sessions using the optimized parameters (no baseline removal, SEF array, features extracted taking the mean of the data windows, and z-scoring the feature vectors). The blue trace corresponds to the saccade BMI decoder performance (left Y axis). The green, red, and black traces correspond to the true negative (no decoder error), true positive (decoder error), and overall DED BMI decoder(err) performance, respectively. The DED BMI decoder(err) performance values correspond to the right Y axis (black axis). The sessions between the dates 121114 and 130415 (YYMMDD) presented more variability and an erratic decoder(err) performance.

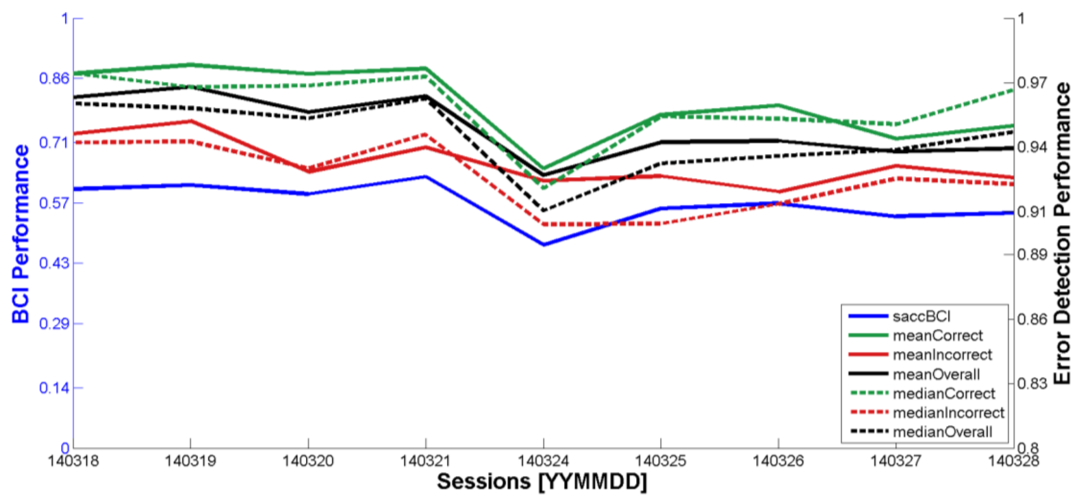


Figure II.27. Decoder(err) performance using 10-fold CV and optimized parameters (monkey J).

Decoder(err) performance across9 monkey J sessions using 10-fold CV and the optimized parameters (no baseline removal, FEF and PFC arrays, features extracted taking the mean of the data windows). The blue trace corresponds to the saccade BMI decoder performance (left Y axis). The green, red, and black traces correspond to the true negative (no decoder error), true positive (decoder error), and overall DED BMI decoder(err) performance, respectively. The DED BMI decoder(err) performance values correspond to the right Y axis (black axis). Continue and dash lines represent the mean and median, respectively, of the 10-fold CV iterations.

3.3.2 Simulating online eLFP detection using different sessions for training and testing. For emulating the training and testing process of the real-time DED BMI sessions, five different rules were considered for selecting the training dataset used by the decoder(err). After training with the dataset specified by each one of the rules, the decoder(err) was tested on all 65 monkey C sessions using its optimized parameters. In Figure II.28 are compiled the results of testing different decoders trained under five rules: the light blue trace represents the *1predSession* rule (using only the previous session for training, and testing on the following session; doing this for each session), the cyan trace represents the *5 predSession* rule (the previous 5 sessions were used for training a decoder that was tested only on the session following the training sessions; doing this for each session),the yellow trace represents the *10 predSessions* rule (the previous 10 sessions were used for training a decoder that was tested only on the session following the training sessions; and repeating this for each session), the red trace represents the *midsession* rule (the first half of all the sessions was used to train a fixed decoder than later was used to test each of the following sessions), the dark blue trace represents the *crrtSession* rule (a decoder was trained and tested using the same session). The saccade BMI performance is included as a reference (black trace). The best decoder(err) performance across sessions was 94.77 ± 1.43 S.D. for the *5predSessions* training rule (cyan trace). The median and quartile values for four of these rules are compiled in Figure II.29. The *5predSession* and *10predSession* approaches provided the best performance values across all 65 sessions.

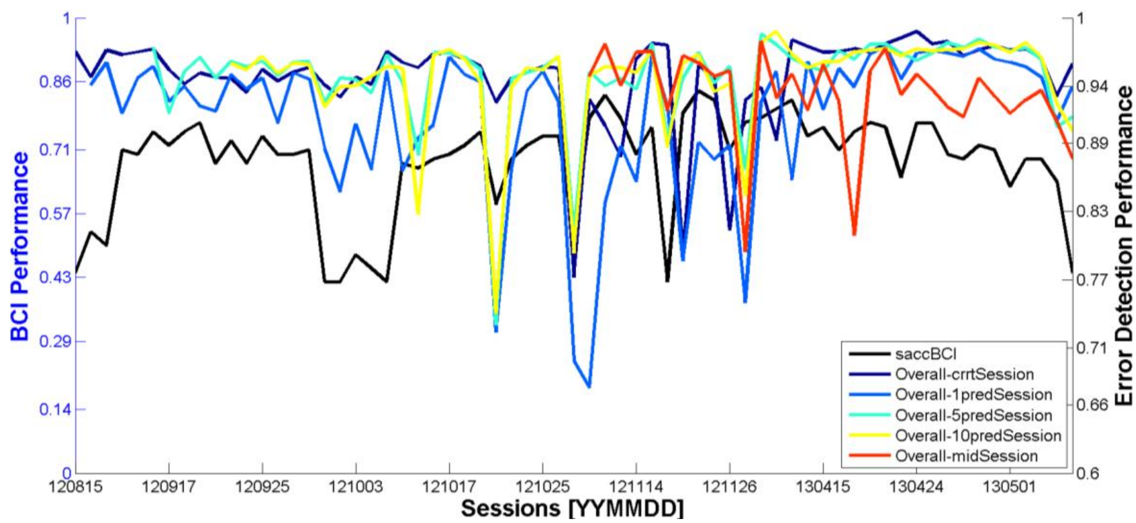


Figure II.28. Decoder(err) performance simulating online eLFP detection using different training datasets (monkey C).

Simulating the training and testing process of an online real-time DED BMI session, the decoder(err) performance was evaluated considering five different rules for selecting the training dataset (employed by the decoder(err)). Saccade BMI performance values are represented by the black trace and the blue Y axis. Decoder(err) performance values correspond to the right Y axis (black axis). The training rules are *crrtSession* (dark blue trace), *1predSession* (light blue trace), *5predSession* (cyan trace), *10predSessio*n (yellow trace), and *midSession* (red trace). For the decoders(err) the optimized parameters from the 10-fold CV analysis was used.

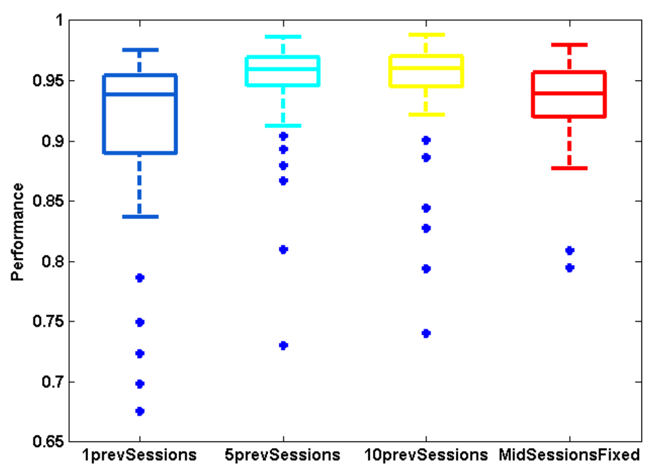


Figure II.29. Boxplots of decoder(err) performance simulating online eLFP detection using different training datasets (monkey C).

The boxplots represent data from simulations of the training and testing processes of an online real-time DED BMI session. The decoder(err) performance was evaluated considering five different rules for selecting the training dataset used by the decoder(err). The training rules are *1predSession* (light blue trace), *5predSession* (cyan trace), *10predSessio*n (yellow trace), *midSession* (red trace) and *crrtSessions* (not shown in the plot). The decoders(err) employed the optimized parameters from the 10-fold CV analysis. The *5predSession* and *10predSession* results provided the best average decoder(err) performance across all 65 monkey C sessions.

3.3 Real-time decoder error detection

The real-time DED BMI sessions took place on two and three consecutive weeks for monkey J and C, respectively (Figure II.30). Before each day's recordings, a decoder(err) was trained following the steps detailed in section 2.5.1. For a list of the sessions used for training the decoder (err) see Table II.3. On each day implementing the decoder error detection BMI, monkey C executed 600 training trials and an average of 536 (±225 S.D.) decoding trials. Monkey J, on the other hand, executed 300 training trials and an average of 812 (±182 S.D.) decoding trials. During this period the saccade BMI made, on average, 157 decoder errors (±63.45 S.D) for monkey C, and 437 decoder errors (±87.52 S.D.) for monkey J. For these weeks the saccade BMI performance was 70,26% (±4.75) and 45.75% (±3.96) for monkey C and monkey J, respectively. The average real-time DED BMI decoder(err) performance was 86.02% (±11.4 S.D) for monkey C, and 87.77 (±7.23 S.D.) for monkey J.

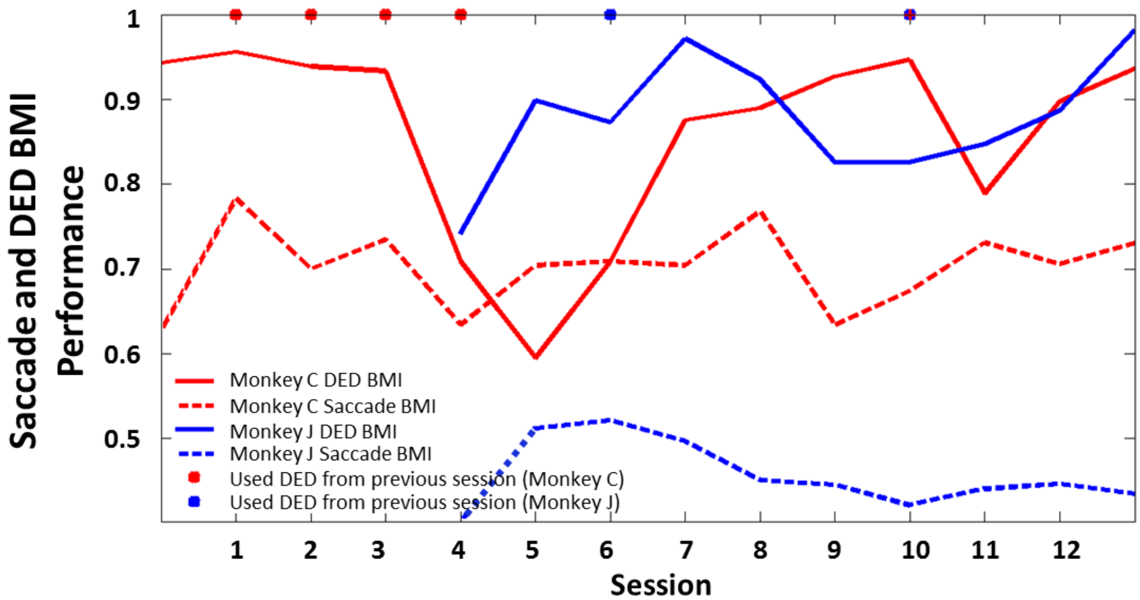


Figure II.30. Real-time DED and saccade BMI decoder performance for both monkeys.

DED and saccade BMI decoder performance for the testing sessions (real-time). Red and blue traces represent values for monkey C and J, respectively. Solid lines indicate DED decoder(err) values, dash lines saccade BMI decoder values. Dots mark sessions in which the decoder(err) was not updated; instead the one from the previous day was used (labeled in the plot as previous sessions decoder). The axis represents the session number. The average decoder(err) performance was 86.02% (±11.4 S.D) for monkey C, and 87.77 (±7.23 S.D.) for monkey J.

3.4 Implant laterality effect on decoder error detection

The implant laterality affected the saccade BMI performance (better performance for the contralateral targets) so this same effect was assessed in the decoder(err) performance (Figure II.31). The ratio of contralateral (and ipsilateral) trials to the total of trials (for each macro subset of trials) was computed. For the contralateral trials, the average ratio for monkey C was 0.487 ± 0.053 S.D (for the correctly classified macro subset) and 0.478 ± 0.146 S.D (for the incorrectly classified macro subset). For monkey J, the average ratio for contralateral trials was 0.492 ± 0.015 S.D (for the correctly classified macro subset) and 0.558 ± 0.079 S.D. (for the incorrectly classified macro subset). Since there was no a significant difference between the ipsilateral and contralateral ratios for monkey C and J, we concluded that there is no a significant difference in the performance of the DED BMI due to the laterality of the implant. This was true for both the correctly and incorrectly classified macro subsets.

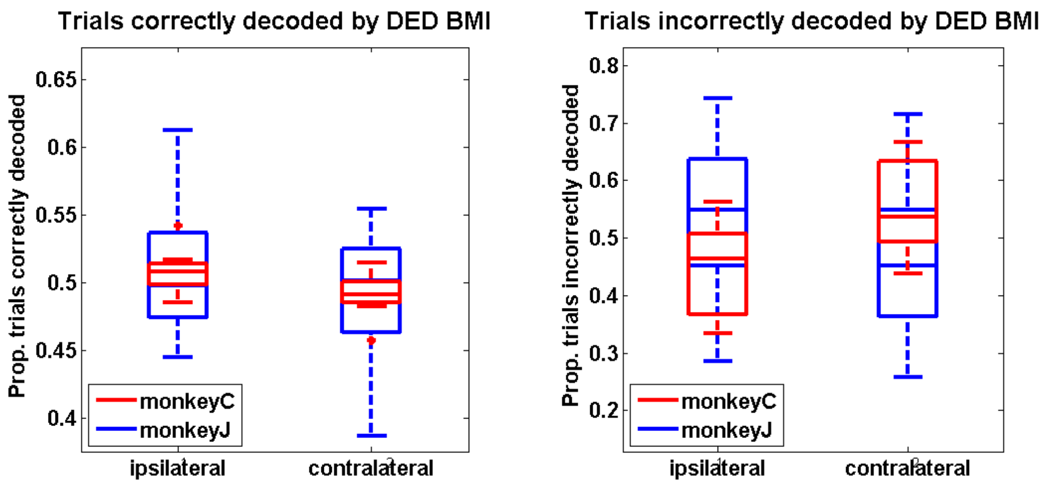


Figure II.31. Ratio of contralateral and ipsilateral trials during DED BMI.

Box plots with the ratio of trials with a specific laterality to the total number of trials (for each decoder(err) outcome condition). In the left plot is the ratio of the number of ipsilateral (left) and contralateral (right) trials to the total number of *correctly* classified trials (trials correctly classified by the DED BMI). In the right plot is the ratio of the number of ipsilateral (left) and contralateral (right) trials to the total number of *incorrectly* classified trials (trials incorrectly classified by the DED BMI). Each box plot presents the median, quartiles, and outliers for monkey C (in red), and monkey J (in blue). In monkey C and J the laterality effect did not significantly affected the decoder(err) performance.

4. Discussion

We presented here, to our knowledge, the first real-time LFP-based DED system integrated within an invasive BMI, and tested on non-human primates. Three major steps were taken to develop the system: describing in detail the decoder error-related LFP, testing the DED system offline, and testing its performance in real-time while two non-human primates (NHP) used a saccade BMI.

A deep analysis of the LFP during BMI control provides a better understanding of the signals, and features, to be used for decoder error detection. The characteristics of

LFP during BMI misclassifications are not reported in the literature so our findings are compared with 1) eLFP recorded during invasive non-BMI tasks and 2) with ErrP signals during non-invasive BMI control. For the time domain features, the small negative trough (100ms) and the positive peak (~200ms) in the SEF array of both monkeys are similar in time and amplitude to results provided by Emeric et al. (2010), who explored eLFP in the SEF of NHPs under a go-no go saccade task. Also, when the ErrDiff trace from eLFP (average incorrect trace minus average correct trace) is compared to traces from EEG BMI studies, our results go along with the shape and timing (although shifted) of the ErrP. Ferrez & Millan (2005) reported times (after the feedback onset) for the negative, positive, and negative peaks of the human ErrP signal (270ms, 350-450ms, 550ms) that are similar to the peaks found in the ErrDiff trace from the eLFP of monkey C (100ms, 100-300ms, 250-500ms, approximately shifted 180ms if compared to Ferrez & Millan results) and monkey J (100ms, 200-300ms, 300-500ms, shifted approximately 157ms if compared to Ferrez & Millan results).

The effect of target location on the eLFP was also studied by Emeric et al. (2010). Those authors found that error signals are related to both incorrect ipsi- and contralateral saccades during the no-go command, supporting our results: data from both monkeys reflect that the amplitude and timing properties apply equally to all 6 target locations without any laterality specificity (although contralateral targets do have larger amplitude peaks in the eLFP signals of monkey C –see Figure II.40, but not in monkey J). Moreover, Emeric et al. suggested eLFP in their SEF recordings represent both error and conflict, agreeing with our distance to true target results. The level of discrepancy (or conflict) significantly modulates the amplitude of the positive peak of the eLFP in the SEF arrays of both monkeys. Also, from data in both the PFC and FEF arrays in monkey J, this level of conflict seems to be significantly modulated by the positive and the secondary (wide) negative peaks found in the ErrP signals. Unfortunately, Emeric et al. only reported data up to 400ms post-saccade or feedback onset (making a comparison with our results impossible), although there is a long list of authors that have linked PFC to response conflict (Ullsperger et al., 2014). The effect of distance to target has been already linked to amplitude modulation (using the positive peak in the signal) in EEG signals during BMI control (Iturrate et al., 2010; Buttfield et al., 2006), supporting our results.

Finally, the effect of previous trial errors found in the late stage of the eLFP (400-500ms post-feedback onset in both monkeys) has behavioral (Godlove et al., 2011; Emeric et al., 2008) and signal correlates (Iturrate et al., 2010; Cavanagh et al., 2010).

Most of the studies focused on eLFP explore the time-domain signal omitting its spectral characteristics. From the time-frequency content found in the error-difference spectrograms in both monkeys, the strongest signatures are an increase in delta, theta, and alpha activity; and a strong decrease in beta power (200-400ms after the feedback onset). The delta, theta and alpha bands were significantly informative of the decoder outcome in our analysis, in line with what we implemented for real-time DED (an LFP filtered in the 1-10Hz band). The beta band was not included in our real-time DED BMI although others include it for decoder error detection (Chavarriaga et al., 2014).

The increases in delta and theta power during error processing are documented in non-BMI situations (Yordanova et al., 2004; Cavanagh et al., 2010; Cohen, 2011) while the increases in alpha and the decreases in beta activity have been previously reported during ErrP BMI control. Chavarriaga et al. (2014) reported that a decrease in beta activity is present when the same users of the BMI have a sense of agency during the BMI task, but not only monitoring errors performed by external agents. In the case of both monkeys, this could mean a possible sense of agency during the saccade BMI control.

Furthermore, an unexpected behavior was found in the high-gamma band in all channels of SEF and FEF arrays, in both monkeys: a decrease of activity lasting 200ms (at 100ms and 300ms post feedback onset for monkey C and monkey J, respectively) immediately followed by an increase in activity with duration of 200ms. A similar decrease of spectral power was found in all the channels in the PFC array of both monkeys, starting at the feedback onset and lasting 100ms and 300ms for monkey J and C, respectively. Increases in high gamma activity during error production have been reported in ECoG data from humans performing a motor task (Milekovic et al., 2012) but the increases last longer (~200-500ms), and without prior decreases in the same frequency band. Increases of high frequency power in LFP during decoder errors have also been reported during BMI control (250ms post-feedback onset), but these results belong to a preliminary study using a single marmoset monkey (Geng et al., 2013) .Upon a closer look at the mean correct spectrogram in monkey J’s PFC channels, a band of power in the high gamma, lasting 100ms, was found right after feedback onset. One possible explanation is that the delivery of the reward by the solenoid added this signal. Nevertheless, the mean incorrect spectrogram (for the same channels, in PFC) also showed an increase in high-gamma (although not as strong as the one in the correct spectrogram), ruling out the effect of the solenoid (reward was never delivered in incorrect trials). The time of the second high gamma transition (in SEF and FEF channels) does not match any behavioral changes or transitions in the task beyond decoder outcome. This high frequency increase needs further analysis. A possible approach is to delay the presentation of the reward to the monkeys so no possible electromagnetic and EMG artifacts can interfere with the signal, hence confirming this high frequency increase is related to incorrect trials.

Beyond the spectrogram analysis, the cross-frequency amplitude-amplitude co-modulation analysis aimed to find possible interactions between alpha and beta rhythms across both monkeys. The coupling results were not consistent across monkeys. In monkey C the cross-frequency amplitude-amplitude coupling analysis showed a decrease in delta-alpha and theta-alpha coupling in the PFC and SEF arrays, and an increase in coupling between different bands for electrodes 18 and 27 (in the PFC array). In monkey J increases and decreases of cross-frequency coupling were present in all the arrays (mainly PFC and FEF). In all three arrays was seen a decrease of alpha-beta, alpha-gamma, and beta-gamma coupling while in PFC and FEF was found an increase in the alpha-delta and alpha-theta coupling. Other methods for evaluating the cross-frequency amplitude coupling should be explored. A logical step is to take into account the dynamics of each trial: instead of computing coupling values for the whole pre- and post-feedback period, calculate it for each time sample across trials and evaluate how significantly it changes over time. A modified version of Voytek et al. (2013) event-related phase-amplitude coupling method could be used for amplitude-amplitude coupling. Moreover, phase-amplitude coupling at each frequency band should also be explored using either Voytek et al. (2013) or Brincat & Miller, (2015) approaches.

The analysis of the eLFP signals allowed testing different features and parameters (offline) to optimize the decode error detection algorithm. The results were specific to each monkey and revealed some variability on the data preprocessing approaches although both provided similar decoder(err) performance of approximately 95% correct classification, proving that the DED BMI design was fast and reliable for single-trial decoder error detection during the closed-loop saccade BMI testing sessions. Finally, the implementation of the DED BMI while the monkeys ran the saccade BMI demonstrated eLFP can be detected in real-time accurately. The overall decoder(err) performance during the real-time sessions was 86.02% (±11.4 S.D) for monkey C, and 87.77 (±7.23 S.D.) for monkey J.

The goal of this chapter was to propose a system employing eLFP for detecting decoder errors, test its performance offline, and implement it in real-time as a module integrated into a running closed-loop BMI. Overall, our findings reveal characteristic eLFP traits that proved to be consistent with the literature and that represent a bridge between the current knowledge about eLFP in non-BMI tasks and ErrP produced during BMI control. Finally, the real-time DED BMI results suggest that decoder error detection can be incorporated as a standard feature in BMI design, which could be tested with adaptive algorithms in order to improve overall decoder performance.

5. Appendix

5.1 Error-related LFP activity per channel and array

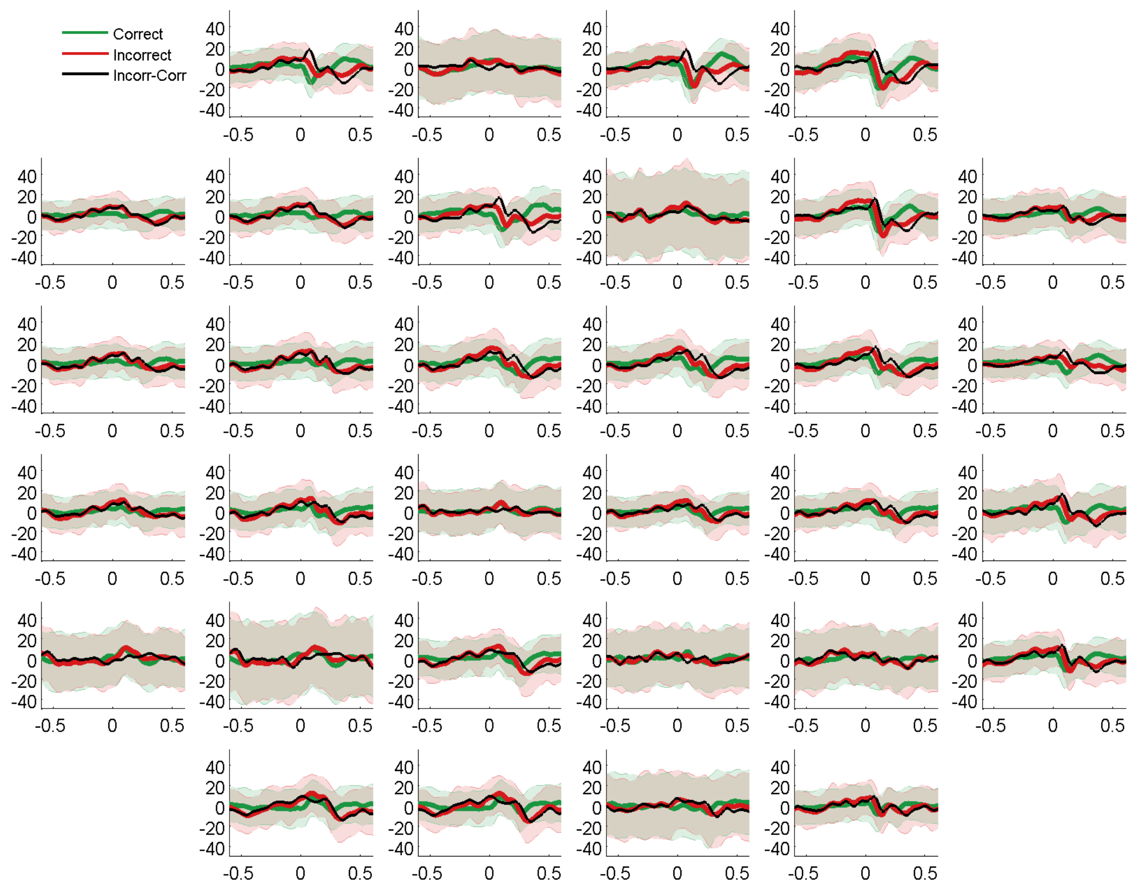


Figure II.32. Average time traces and S.D. for all channels in the PFC array of an example session (monkey C).

From a representative session from monkey C we plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the PFC array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

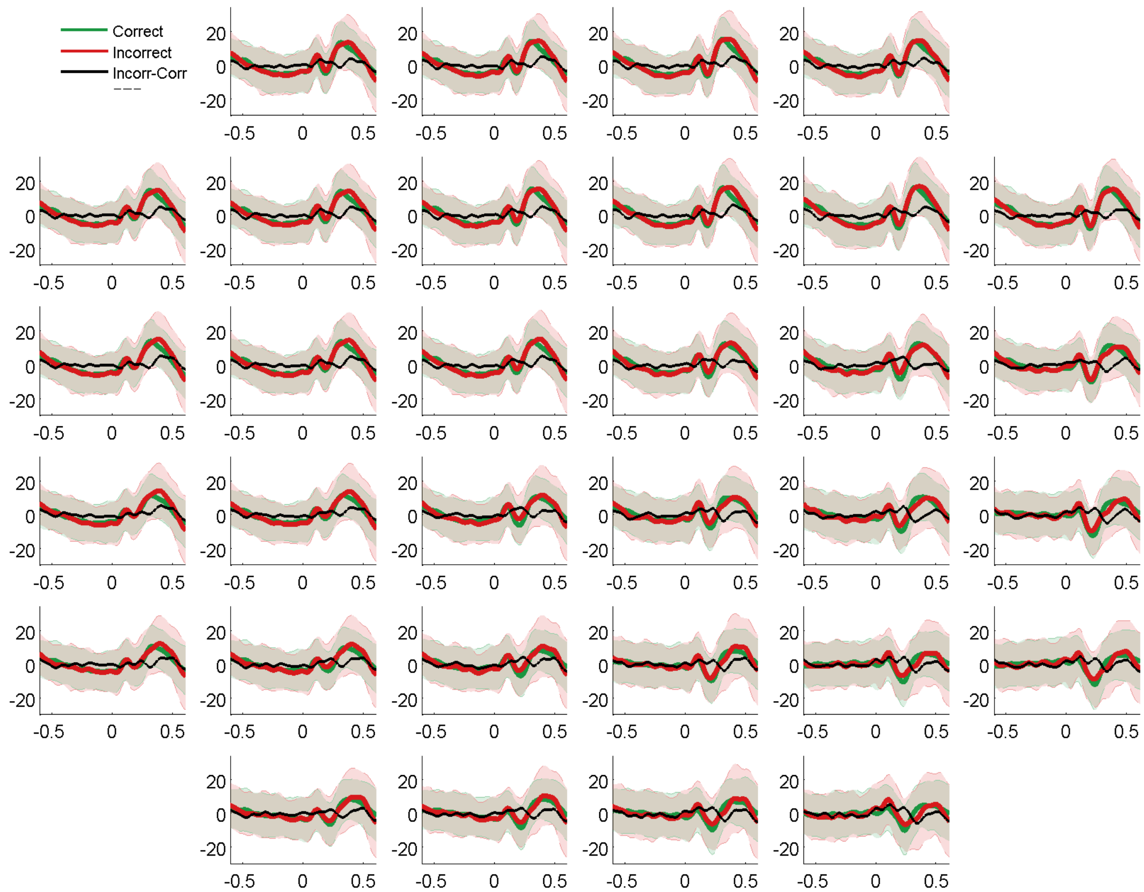


Figure II.33. Average time traces and S.D. for all channels in the FEF array of an example session (monkey C).

From a representative session from monkey C we plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the FEF array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (µV), the X axis time (seconds with respect to feedback onset time).

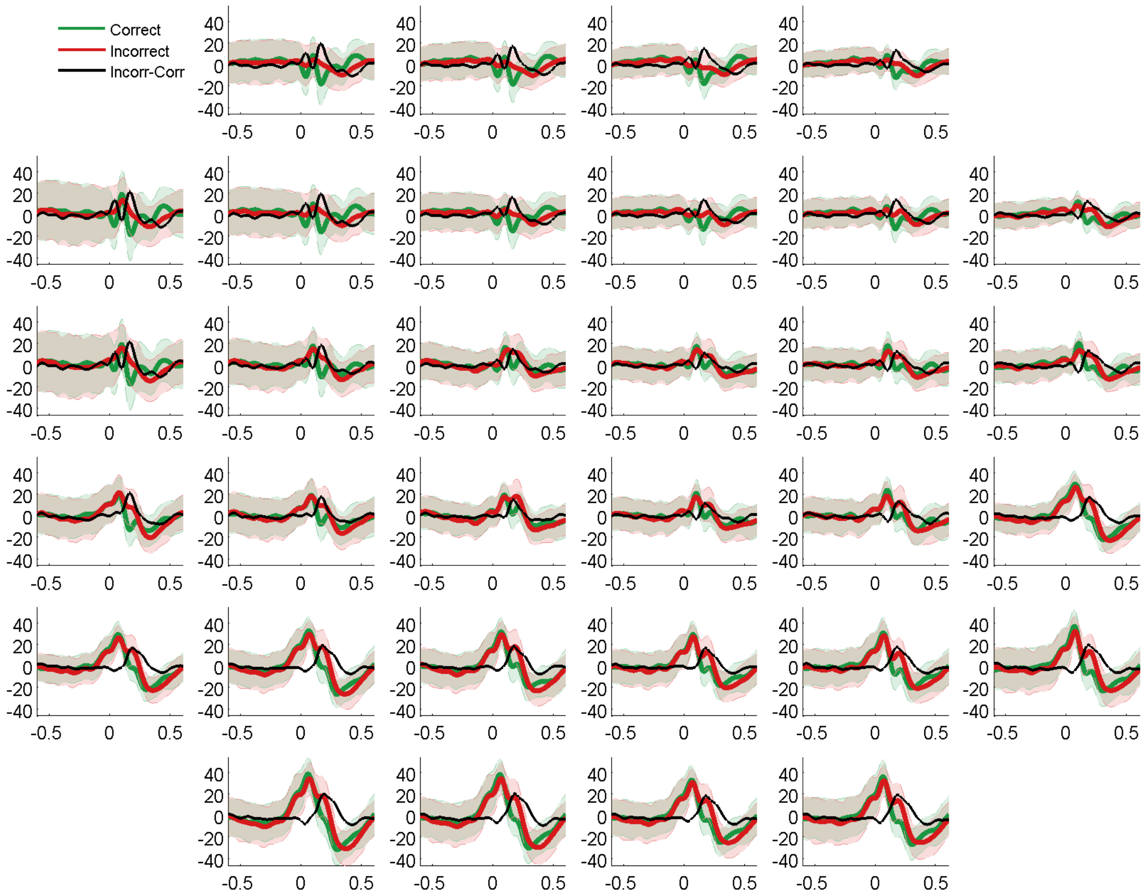


Figure II.34. Average time traces and S.D. for all channels in the SEF array of an example session (monkey J).

From a representative session from monkey J we plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the SEF array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

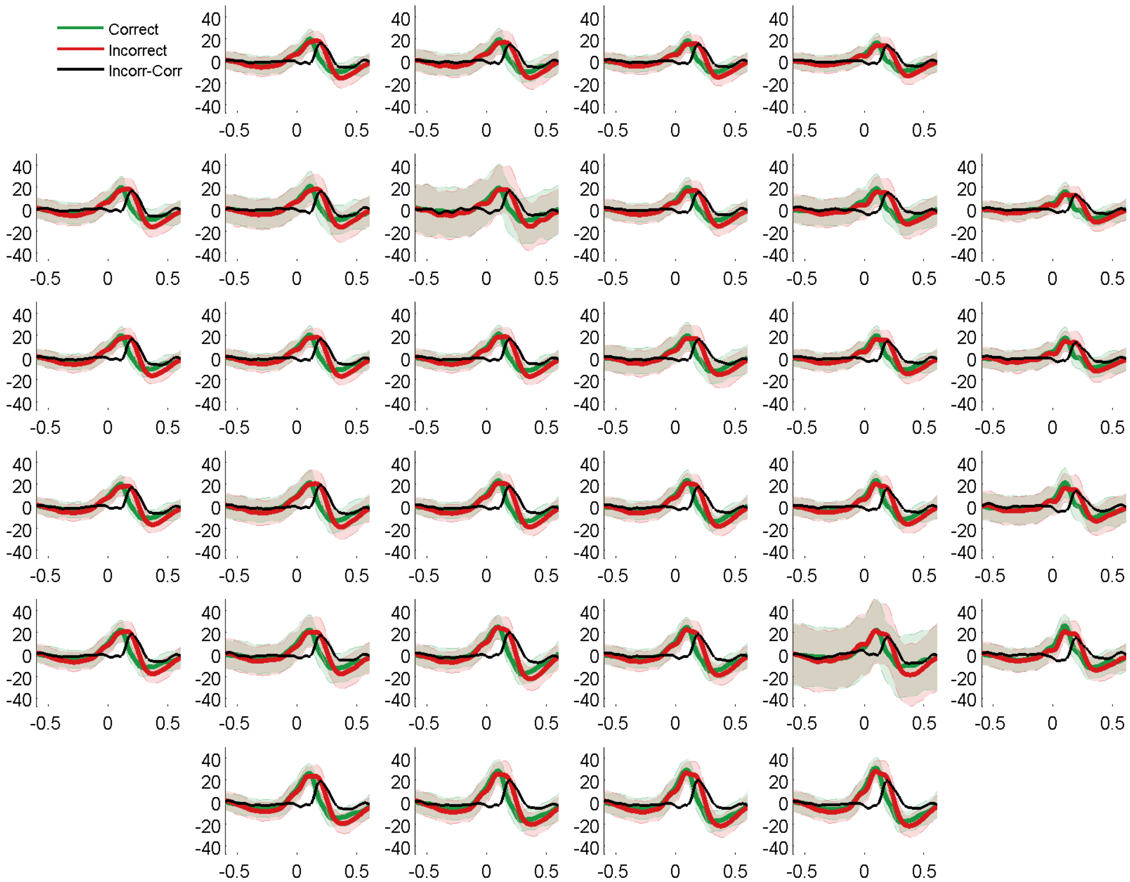


Figure II.35. Average time traces and S.D. for all channels in the FEF array of an example session (monkey J).

From a representative session from monkey J we plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the FEF array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

5.2 Error-related LFP for each true target location

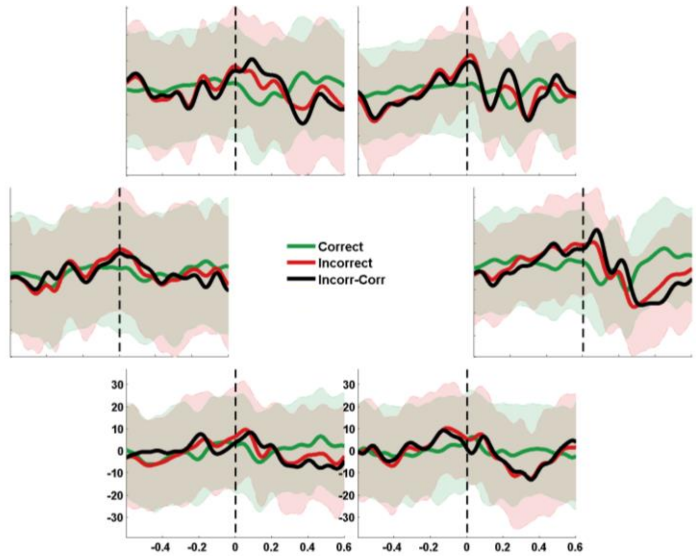


Figure II.36. PFC array mean time traces and S.D. for each target location (monkey C).

Mean time traces and S.D. taken from all epochs and channels from a representative monkey C session for each target location. We plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the PFC array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

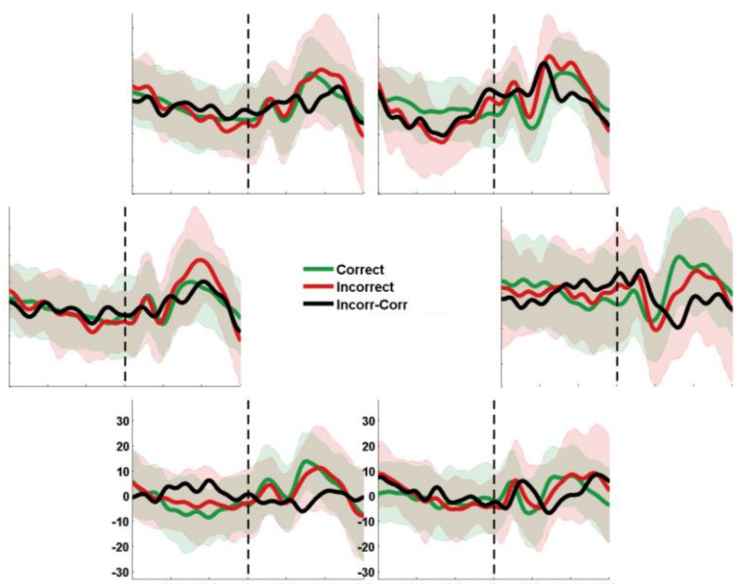


Figure II.37. FEF array mean time traces and S.D. for each target location (monkey C).

Mean time traces and S.D. taken from all epochs and channels from a representative monkey C session for each target location. We plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the FEF array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

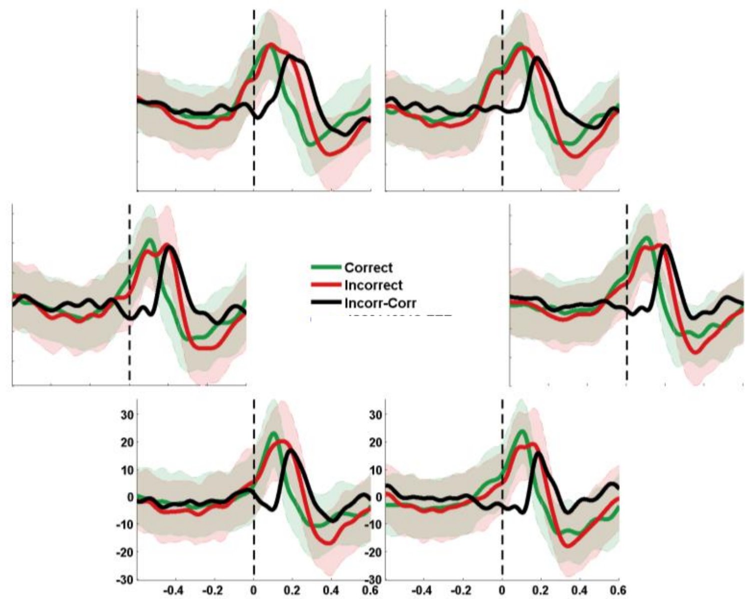


Figure II.38. FEF array mean time traces and S.D. for each target location (monkey J).

Mean time traces and S.D. taken from all epochs and channels from a representative monkey J session for each target location. We plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the FEF array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

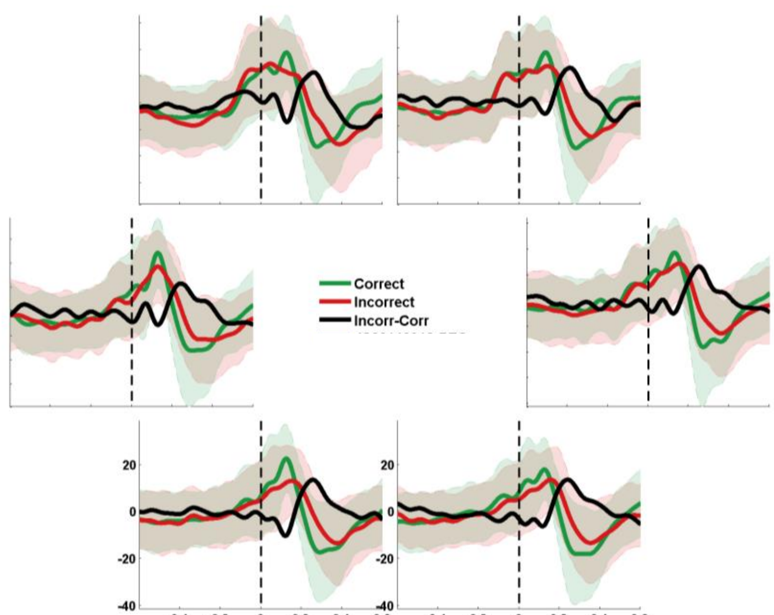


Figure II.39. PFC array mean time traces and S.D. for each target location (monkey J).

Mean time traces and S.D. taken from all epochs and channels from a representative monkey J session for each target location. We plot the correct (green trace), incorrect (red trace), and ErrDiff (black trace) mean traces for the 32 channels in the PFC array. The epochs' S.D. is presented as error bars. Y axis represents the voltage (μV), the X axis time (seconds with respect to feedback onset time).

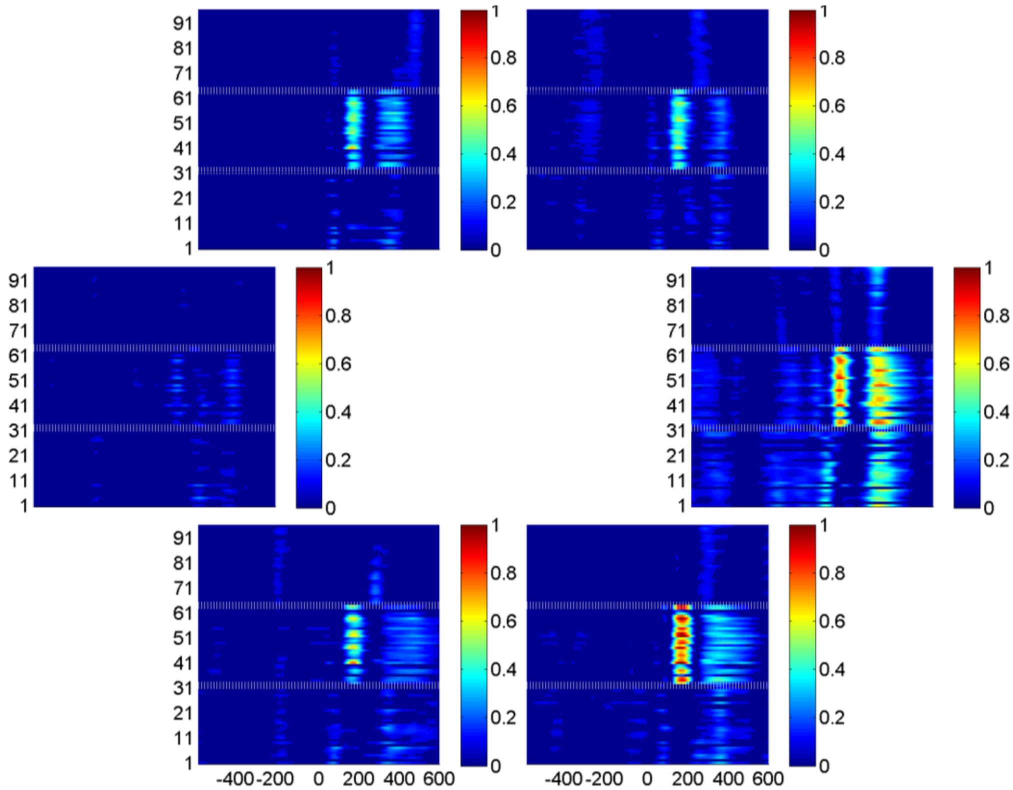


Figure II.40. Normalized LFP explained variance (across target location) given the decoder outcome for each target location and array (monkey C).

Normalized (across target location) LFP explained variance due to decoder outcome for monkey C. The Y axis represents electrode (channel) number and the X axis the time to feedback onset (milliseconds). Electrodes 1-32 are for PFC, 33-64 for SEF, and 65-96 for FEF. For all target locations, SEF electrodes, in the range 100-200ms, show significant decoder outcome information (two-sample T-test $p < 0.01$). For targets 1, 2, 3, 6 (with targets 1, 2, and 6 being contralateral to the implants) the same is true in SEF in the range 250-450ms. Targets 1, 2, 5, and 6 present significant information given the decoder outcome (250-450ms) at some PFC electrodes. The colorbar represents the explained variance. Ipsilateral targets (3, 4, and 5) show less significant explained variance for decoder outcome.

5.3 Error-related LFP distance to true target

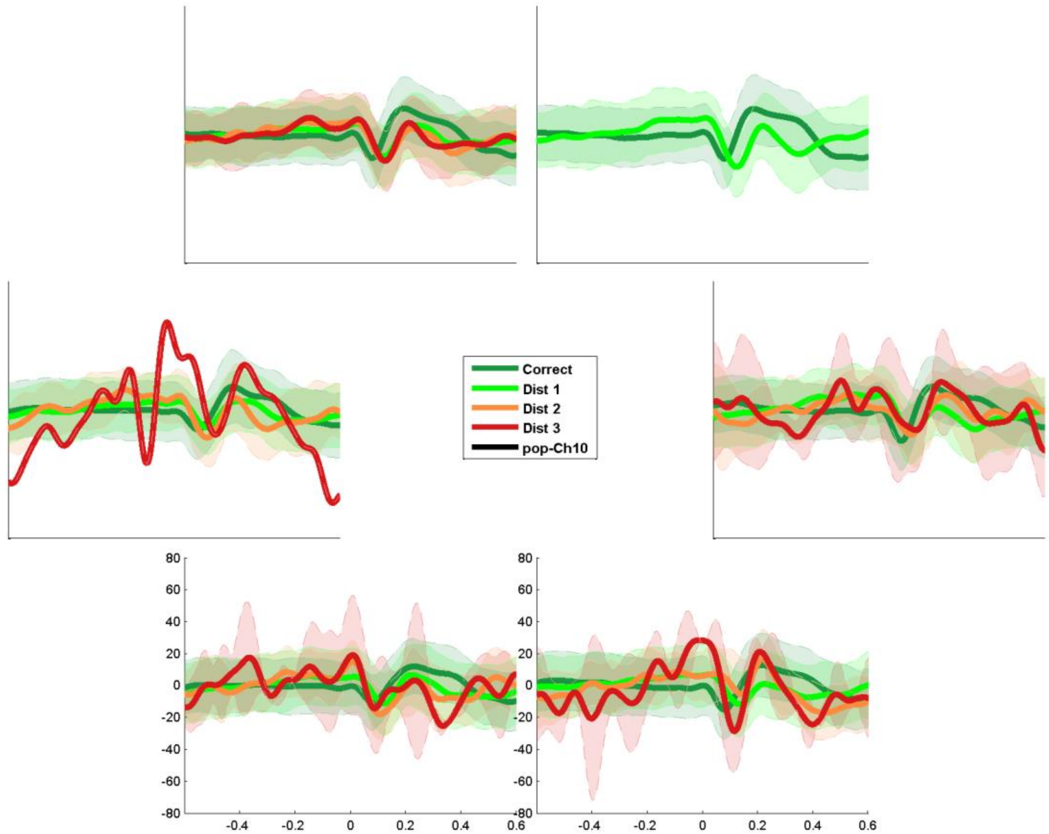


Figure II.41. Population distance to true target for each target location in one PFC channel (monkey C).

Mean distance to true target (across all epochs from a population of monkey C sessions) for each target location at one PFC channel. Dark green, light green, orange, and red traces represent the correct, dist1, dist2, and dist3 mean traces, respectively. For most of the targets the positive peak amplitude seems to be modulated by the distance to true target: larger amplitude for larger distance to true target Error bars represent the epochs' standard deviation.

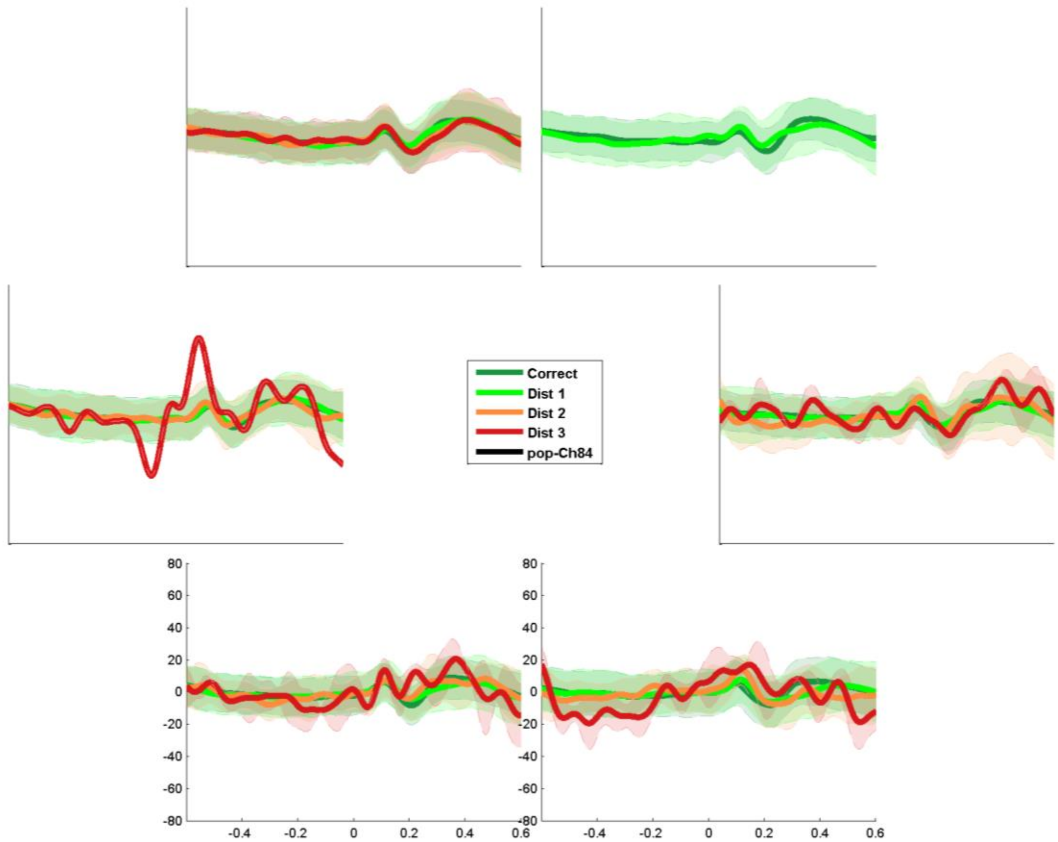


Figure II.42. Population distance to true target for each target location in one FEF channel (monkey C).

Mean distance to true target (across all epochs from a population of monkey C sessions) for each target location at one FEF channel. Dark green, light green, orange, and red traces represent the correct, dist1, dist2, and dist3 mean traces, respectively. For most of the targets the positive peak amplitude seems to be modulated by the distance to true target: larger amplitude for larger distance to true target Error bars represent the epochs' standard deviation.

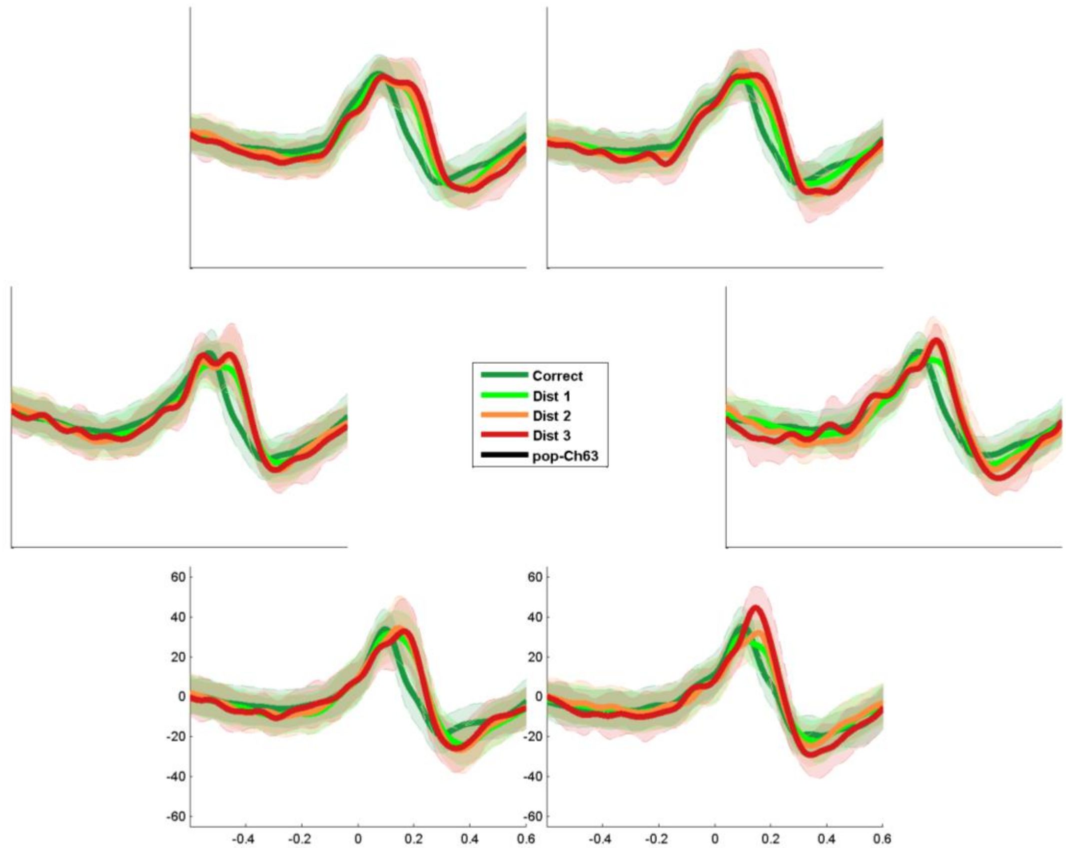


Figure II.43. Population distance to true target for each target location in 1 FEF channel (monkey J).

Mean distance to true target (across all epochs from a population of monkey C sessions) for each target location at one FEF channel. The Y axis represents the signal amplitude (μV) and the X axis the time to feedback onset (seconds). Dark green, light green, orange, and red traces represent the correct, dist1, dist2, and dist3 mean traces, respectively. For most of the targets the positive peak amplitude seems to be modulated by the distance to true target: larger amplitude for larger distance to true target Error bars represent the epochs' standard deviation.

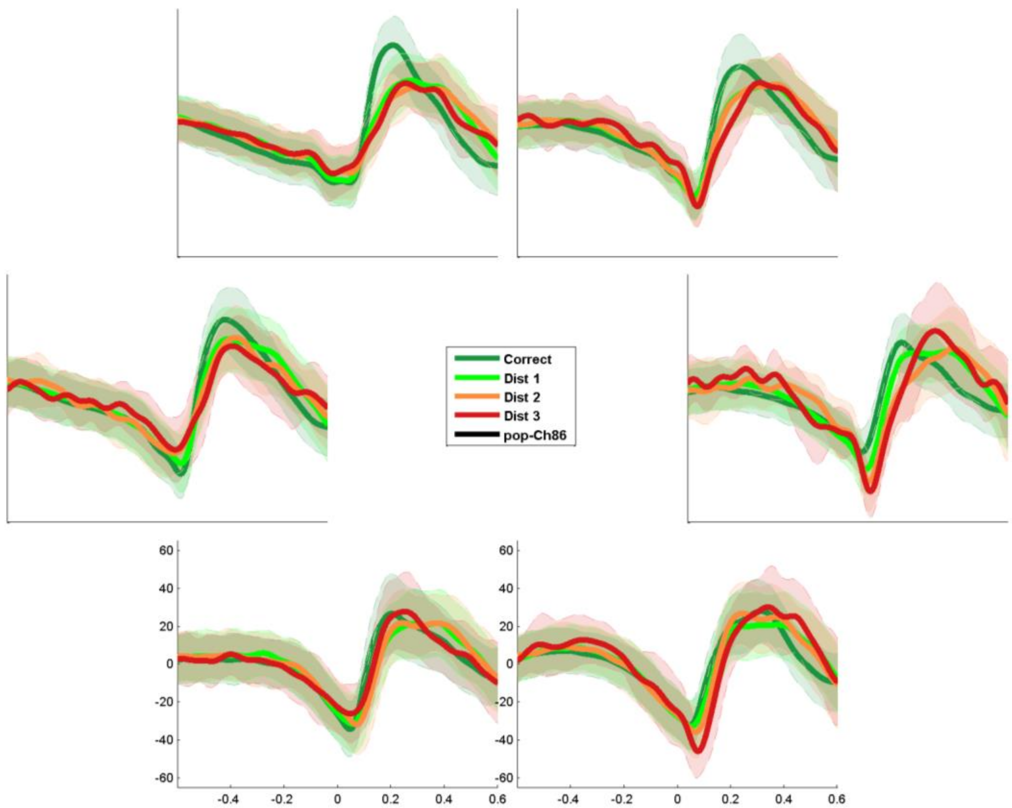


Figure II.44. Population distance to true target for each target location in 1 PFC channel (monkey J).

Mean distance to true target (across all epochs from a population of monkey C sessions) for each target location at one PFC channel. The Y axis represents the signal amplitude (μV) and the X axis the time to feedback onset (seconds). Dark green, light green, orange, and red traces represent the correct, dist1, dist2, and dist3 mean traces, respectively. For most of the targets the positive peak amplitude seems to be modulated by the distance to true target: larger amplitude for larger distance to true target Error bars represent the epochs' standard deviation.

5.4 Log difference power spectrograms

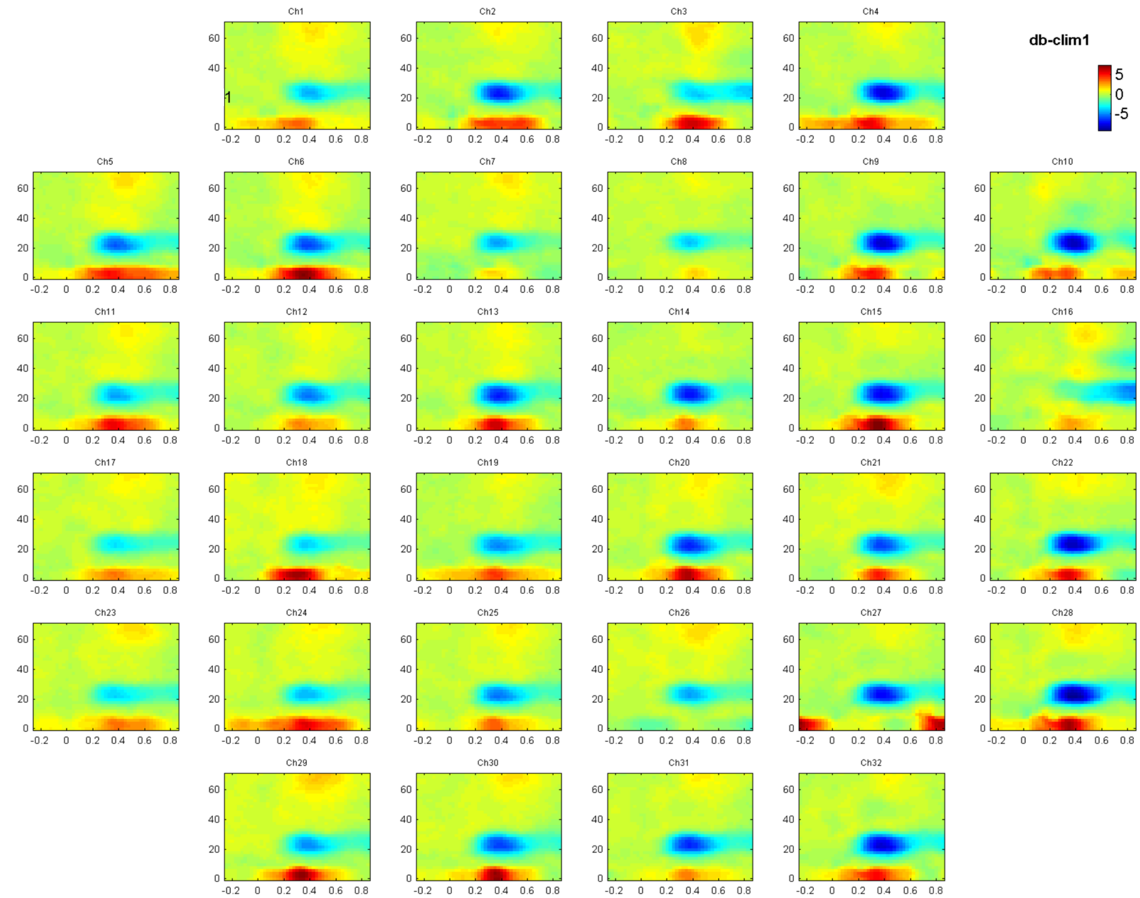


Figure II.45. Spectrogram log power difference for PFC array in monkey C.

ErrDiff spectrogram (average correct spectrogram subtracted from the average incorrect spectrogram) of a population of monkey C sessions for all channels in the PFC array in the 0-70Hz frequency band. All channels are equally scaled and the power, in decibels (dB), is represented in the color bar.

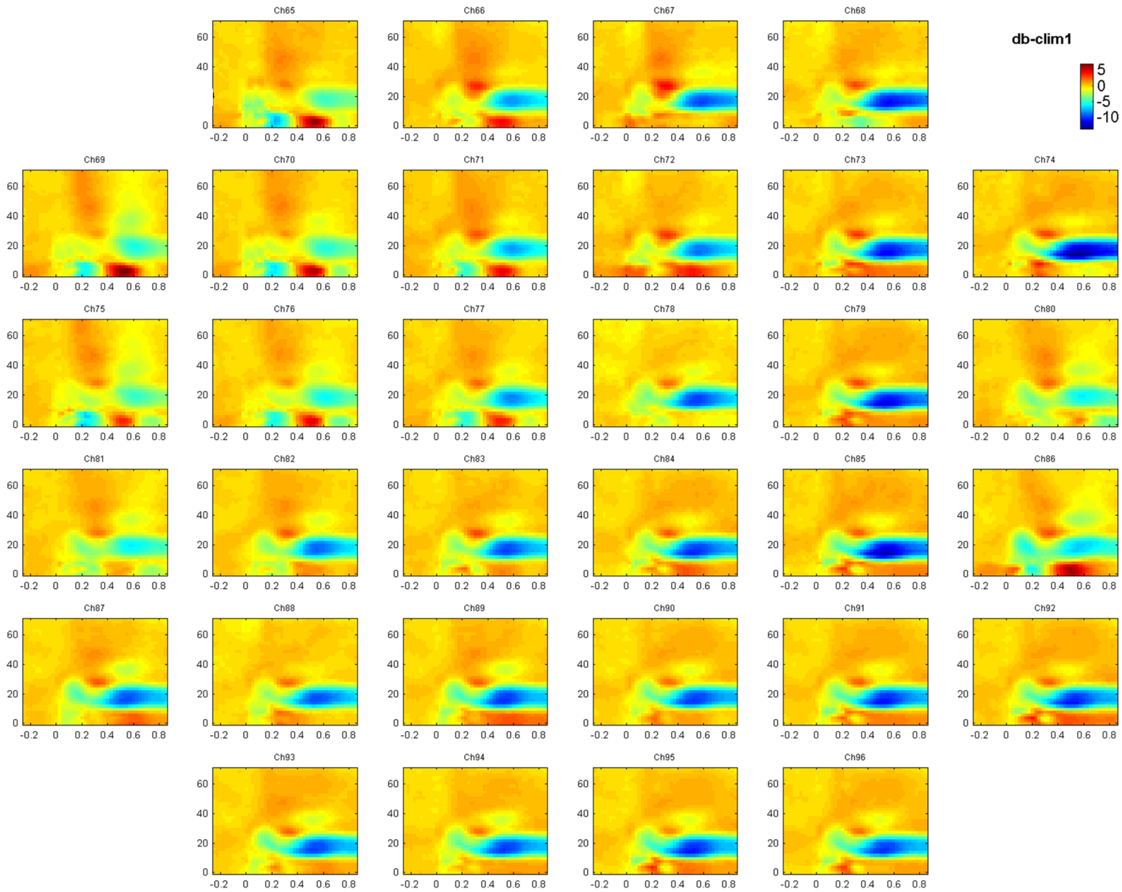


Figure II.46. Spectrogram log power difference for PFC array in monkey J.

ErrDiff spectrogram (average correct spectrogram subtracted from the average incorrect spectrogram) of a population of monkey J sessions for all channels in the PFC array in the 0-70Hz frequency band. All channels are equally scaled and the power, in decibels (dB), is represented in the color bar.

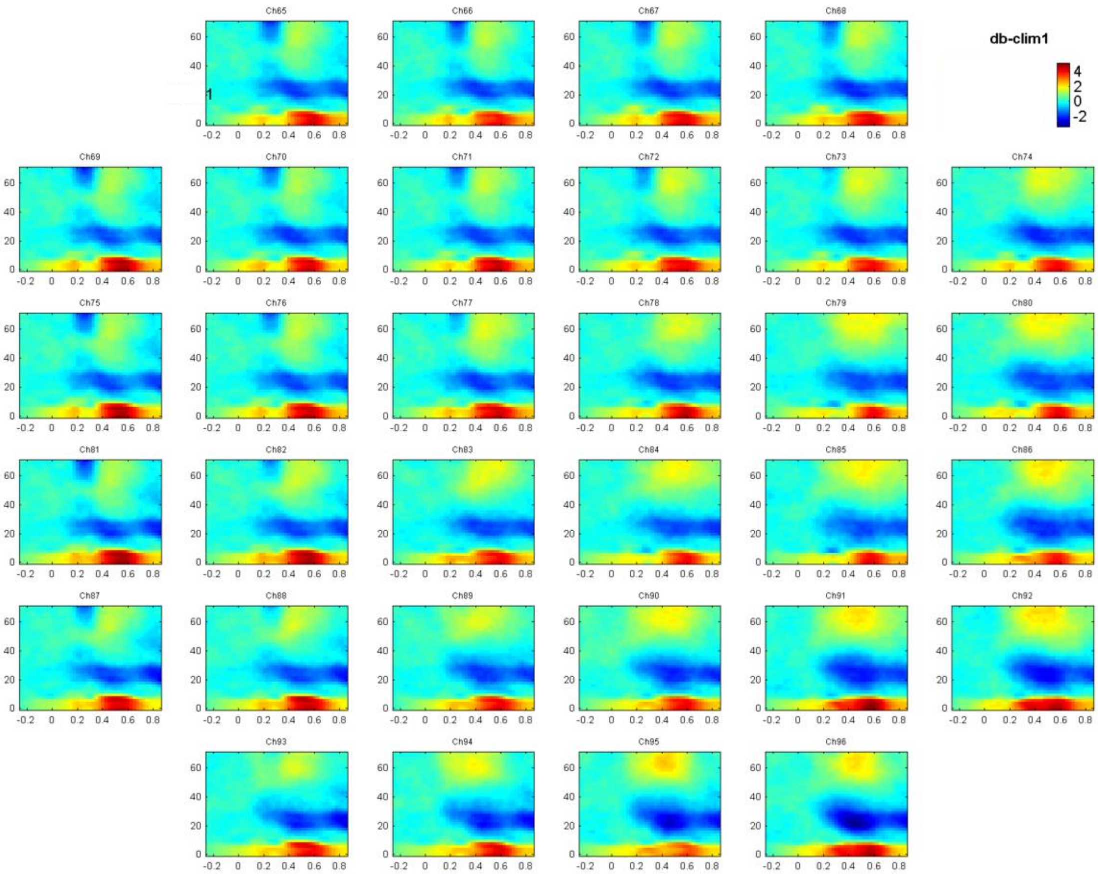


Figure II.47. Spectrogram log power difference for FEF array in monkey C.

ErrDiff spectrogram (average correct spectrogram subtracted from the average incorrect spectrogram) of a population of monkey C sessions for all channels in the FEF array in the 0-70Hz frequency band. All channels are equally scaled and the power, in decibels (dB), is represented in the color bar.

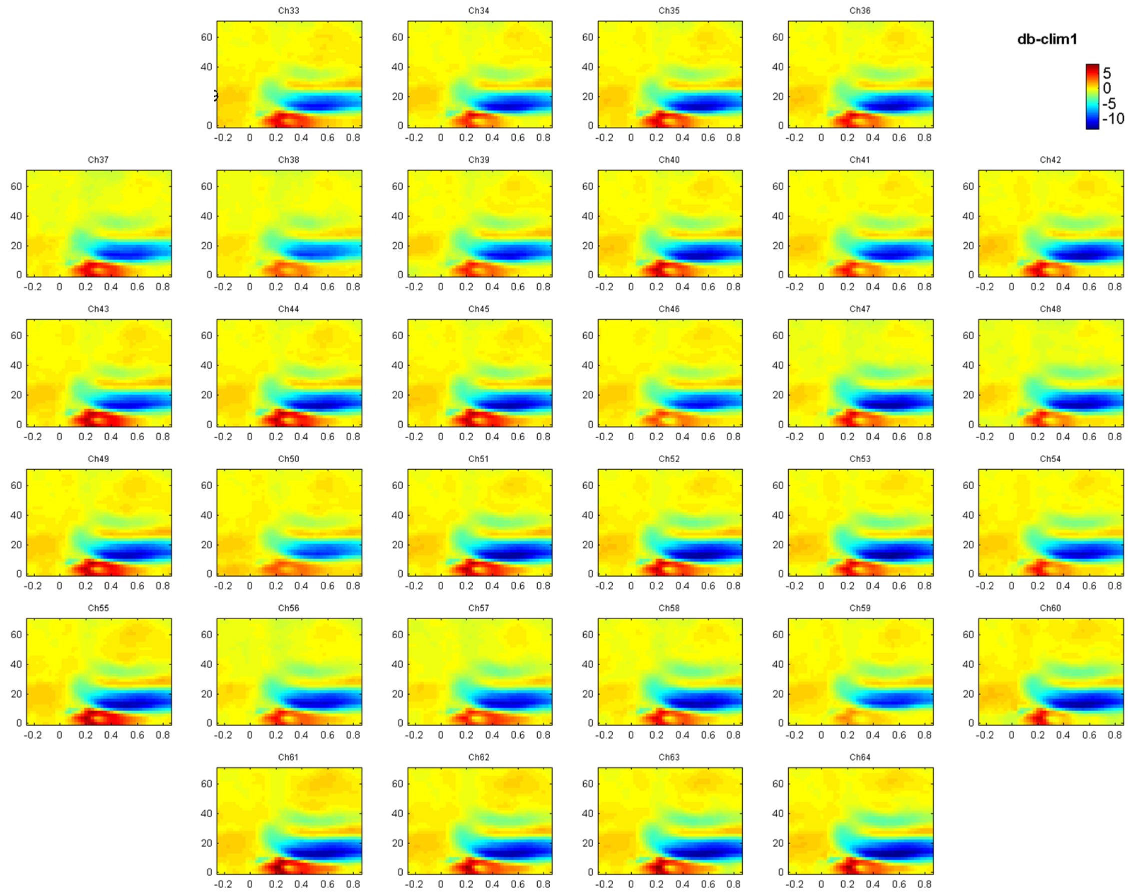


Figure II.48. Spectrogram log power difference for FEF array in monkey J.

ErrDiff spectrogram (average correct spectrogram subtracted from the average incorrect spectrogram) of a population of monkey J sessions for all channels in the FEF array in the 0-70Hz frequency band. All channels are equally scaled and the power, in decibels (dB), is represented in the color bar.

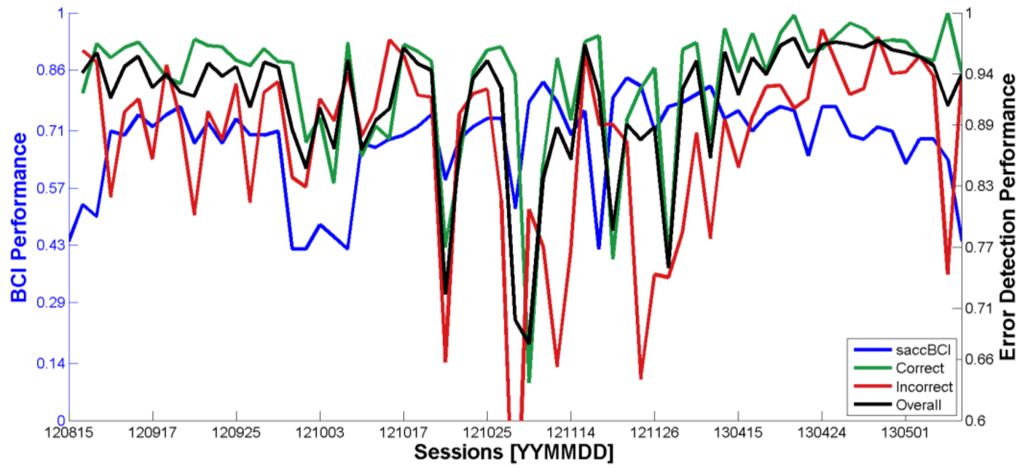


Figure II.49. Decoder(err) performance simulating online ErrP detection using the previous day sessions for training (monkey C).

Simulating the training and testing process of an online real-time DED BMI session, we evaluated the decoder(err) performance using previously recorded data (1 sessions), and testing it on the new day's data. These traces correspond to the training rule *1predSessions*. The analysis was performed across 65 monkey C sessions using the optimized parameters (no baseline removal, SEF array, features extracted taking the mean of the data windows, and z-scoring the feature vectors). The blue trace corresponds to the saccade BMI decoder performance (left Y axis). The green, red, and black traces correspond to the true negative (no decoder error), true positive (decoder error), and overall DED BMI decoder(err) performance, respectively. The DED BMI decoder(err) performance values correspond to the right Y axis (black axis).

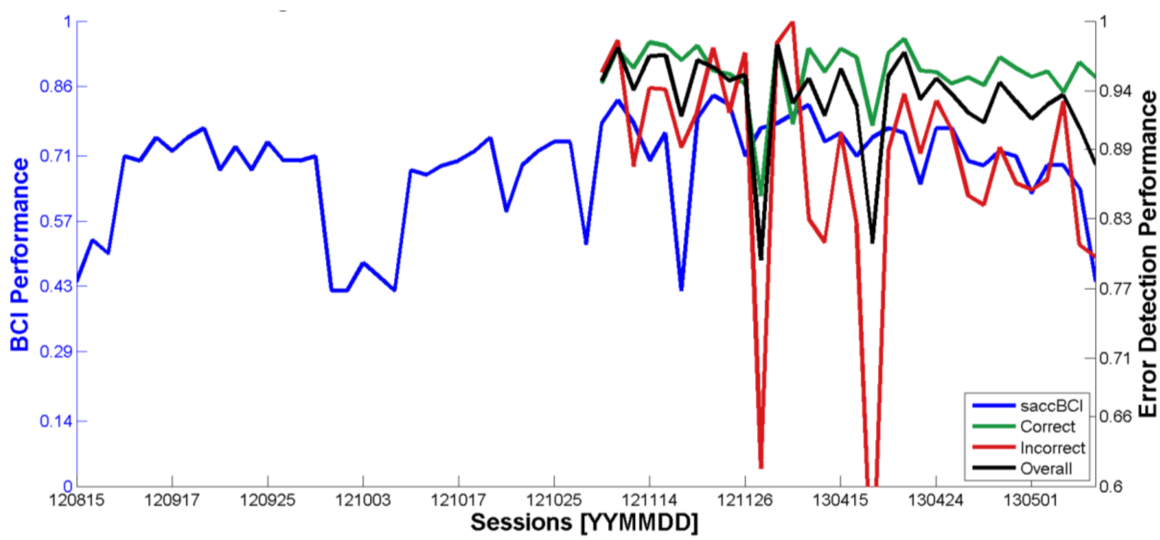


Figure II.50. Decoder(err) performance simulating online ErrP detection with a fixed decoder trained on 33 sessions training (monkey C).

Decoder(err) performance using a fixed decoder(err) trained on the first 33 sessions from the 65 sessions dataset. These traces correspond to the training rule *MidSession.* The decoder(err) used the optimized parameters (no baseline removal, SEF array, features extracted taking the mean of the data windows, and z-scoring the feature vectors). The blue trace corresponds to the saccade BMI decoder performance (left Y axis). The green, red, and black traces correspond to the true negative (no decoder error), true positive (decoder error), and overall DED BMI decoder(err) performance, respectively. The decoder(err) performance values belong to the right Y axis (black axis).

5.6 Online real-time eLFP detection

Table II.3. Sessions used for training the online real-time DED BMI for monkey C.

Here we present a summary of the first and last sessions used for training the decoder run on each real-time DED BMI sessions. The type of decoder column signals if the decoder was new or repeated. Repeated decoders were used when new data was not available so the previous one was used instead.

Session No.	Monkey	First and last session used for training	Total # sessions	Type of decoder
1	C	CS20140303-CS20140328-12	12	New
2	C	CS20140303-CS20140328-12	12	*Repeated*
3	C	CS20140303-CS20140328-12	12	*Repeated*
4	C	CS20140303-CS20140328-12	12	*Repeated*
5	C	CS20140303-CS20140328-12	12	*Repeated*
6	C	CS20140303-CS20140414-16	16	New
7	C	CS20140303-CS20140411-15	15	New
8	C	CS20140415	1	New
9	C	CS20140409-CS20140417-7	7	New
10	C	CS20140409-CS20140418-8	8	New
11	C	CS20140409-CS20140418-8	8	*Repeated*
12	C	CS20140409-CS20140421-9	9	New
13	C	CS20140409-CS20140423-11	11	New
14	C	CS20140421-CS20140424-4	4	New
1	J	JS20140328-JS20140318-9	9	New
2	J	JS20140318-JS20140328-9	9	New
3	J	JS20140318-JS20140328-9	9	*Repeated*
4	J	JS20140415-JS20140416-2	2	New

Session No.	Monkey	First and last session used for training	Total # sessions	Type of decoder
5	J	JS20140414-JS20140417-4	4	New
6	J	JS20140414-JS20140418-5	5	New
7	J	JS20140414-JS20140418-5	5	*Repeated*
8	J	JS20140414-JS20140421-6	6	New
9	J	JS20140414-JS20140423-8	8	New
10	J	JS20140421-JS20140424-4	4	New

CHAPTER III: ERROR-RELATED POTENTIALS FOR DIRECT VOLITIONAL CONTROL AND HUMAN-ROBOT INTERACTION

1. Introduction

Communicating with a robot using electroencephalography (EEG) signals via a brain-machine interface (BMI) could provide a direct and fast feedback loop that is easy and natural for humans, thereby enabling a wide variety of intuitive interaction tasks between humans and robots. Moreover, the use of these signals for control could offer exciting possibilities for intuitive human-robot interaction (HRI), especially for people with motor disabilities. For this community, HRI could be used to ambulate, interact with the world, and perform daily tasks (such as feeding and cleaning) via the control of an external system such as a robotic agent.

The emergence of non-invasive EEG and its natural applicability to human-robot collaboration tasks has spurred much research over the past decades. EEG signals have been used with motor imagery tasks to control robots such as a quadcopter or wheelchair (LaFleur et al., 2013; Iturrate et al., 2009; Tonin et al., 2011). While these certainly showcase the promise of using EEG for robot applications, typical approaches often require several training phases to pre-screen operators based on task proficiency and for the human to learn how to modulate the brain activity appropriately (Guger et al., 2012). During BMI control, the most common signals produced in response to stimuli (usually visual) are the P300, in conventional grids (Farwell & Donchin, 1988; Hoffmann et al., 2006) or during rapid serial visual presentation (RSVP, Acqualagna et al., 2010), and the steady state visually-evoked potentials (SSVEP, Guger et al., 2012; Ramli et al., 2015).

While these signals have shown good average classification performance, they require constant attention, add additional cognitive or visual burden, and usually require many repeated prompts in order to decode a single command. Such challenges make these approaches less amenable to closed-loop control tasks.

For HRI, a more desirable approach would be one that utilizes naturally occurring brain activity. This would allow for generalization to many different tasks and would not require extensive training, active modulation from the user, or a high cognitive load. A possible candidate is the error-related potential (ErrP), initially described by Falkenstein et al. (1991) and Gehring et al. (1993). This evoked-potential is consistently generated when an observer consciously or unconsciously recognizes that an error has been committed, even if the error is made by someone or something else (Chavarriaga & Millan, 2010; Chavarriaga et al., 2014; Iturrate et al., 2010; Ferrez & Millan, 2005). ErrP signals reflect part of the cognitive processes involved in performance monitoring and attention modulation. They also present characteristic properties and characteristic timing that exist across users with no prior training, regardless of the task (Ullsperger et al., 2014; Chavarriaga et al., 2014). In addition, these signals are typically detectable within 500ms from the feedback onset time (Chavarriaga & Millan, 2010), suggesting ErrP are particularly well-suited for fast HRI. Recently, efforts have been made to apply ErrP to robotic tasks (Lopez-Larraz et al., 2010; Iturrate et al., 2014). Although ErrP have been shown to be useful for improving decoder performance via reinforcement learning (Spüler et al., 2012), this approach has not been used in closed-loop tasks where this is the sole control signal. More recently, ErrP have been found in more complex tasks, such as semi-autonomously navigating a wheelchair (Perrin et al., 2010), and have been classified in online settings (in a real-life driving task, Zhang et al., 2015). Nevertheless, no closed-loop feedback has been included in any of these systems. Due to the inherent difficulty of quickly extracting ErrP from a subject's brain activity, studies involving error-related potentials are often performed in controlled settings and for simulated or open-loop tasks. However, robotic applications demand closed-loop scenarios in real-world environments.

In this chapter we propose and implement a BMI paradigm that relies only on non-invasive ErrP for direct binary-choice decoding. Reliably detecting ErrP signals could enable communication via a signal that occurs naturally in the brain during interaction with, or observation of, a collaborating robot. This could potentially alleviate extensive user training, extra cognitive load, or constant visual stimuli often required by BMI. Specifically, we explore the applicability of ErrP signals during observation (oErrP) and interaction (iErrP) to real-time closed-loop robotic tasks in a noisy environment, and the effect of the closed-loop on these signals. Towards this end, a feedback system was developed for human-robot collaboration. In particular, a Rethink Robotics Baxter robot was tasked with object selection while being observed by a human (from the general population and who has not been trained or screened on BMI control proficiency). The human operator's EEG signals were collected and decoded in real-time during the task (while Baxter randomly selected one of the targets); if an ErrP signal was detected, the robot immediately corrected its trajectory (providing closed-loop feedback to the user).

Previous studies using EEG data rely on averaging hundreds of trials in order to characterize and detect specific signal properties given the low signal-to-noise ratio (SNR) of the recorded data. In this chapter, we use trial averages only for characterizing the ErrP but use single-trial data when performing offline and closed-loop ErrP detection.

In the Methods section of this chapter we present the design and implementation of the complete system as well as the different offline analyses performed to optimize the parameters implemented in the closed-loop sessions. The Results section shows the classification performance values for open-loop and closed-loop experiments as well as offline analysis of both oErrP and iErrP. Offline performance values were better than those obtained in the closed-loop online system. Interaction ErrP signals were found to be easier to classify than observation ErrP, opening new opportunities for implementing both ErrP detection systems to improve closed-loop decoder performance. Overall, this work demonstrates the potential of ErrP for seamless robotic control via direct volitional control, and moves closer towards the goal of real-time intuitive human-robot interaction.

2. Methods

To explore the use of ErrP for direct volitional control, two different paradigms were implemented and tested: 1) we presented still images representing a robot arm selection and 2) used an actual robot performing a selection. In both cases subjects were asked to evaluate if the selection was correct or not. Different analyses were performed with the data collected in each paradigm. The binary-choice still images paradigm was used as a proof of concept that later was expanded in the binary-choice reaching paradigm. The proof of concept was required due to the complexity of the binary-choice reaching paradigm. Several technical problems (due to the different subsystems required for closed-loop decoding) were solved before implementing the reaching task.

2.1 Subject selection

All subjects provided informed consent for the study, which was approved by the Internal Review Board of Boston University and the Committee on the Use of Humans as Experimental Subjects of MIT. For all EEG recordings, participants were recruited through community advertisements at Boston University and MIT, were selected from the general population and did not undergo any training sessions (for BMI control). Subjects were not screened based on their innate ability to produce the desired ErrP or their experience with EEG or BMI. For the still images task, 6 subjects were recruited (83.33% right-handed, 83.33% male). For the open-loop reaching task experiments, we collected data from 8 subjects in 11 sessions (three subjects repeated their participation and the probability of Baxter choosing the correct option was 70% in these last three sessions). From the population participating in the open-loop reaching task, 90.91% were right-handed and 63.33% male. For the closed-loop sessions, we divided the population according to naïve or trained subjects. Two subjects from previous recordings (subjects 001 and 016) participated in 5 closed-loop sessions (100% right-handed, 50% male). Finally, five naïve subjects participated in the closed-loop reaching sessions (100% right-handed, 100% male).

From the group of naïve subjects, one of them participated while in a deep meditative state (subject M). The Baxter robot was located between a hallway (with heavy pedestrian traffic) and an entrance door, making this area very noisy and distracting. To avoid such distractions, the subjects wore earplugs and Baxter's working area was surrounded by black curtains. Nevertheless, these measures were not 100% effective, and the subjects struggled to stay focused on the task. Based on this experience, we wanted to evaluate the effect that meditation had on the performance of the task.

Our rationale stemmed from the idea that during meditation a subject would be able to focus his attention in the task and filter out any distractions from the external world.

2.2 Binary-choice still images paradigm

We designed and implemented a binary-choice task using still images of Baxter reaching to two different targets. This paradigm allowed reliable timing of the events, including the feedback onset time, and served as a proof of concept before moving to a more complex scenario, such as the one involving real robot arm reaches.

2.2.1 Stimulus presentation. PsychToolbox for Matlab (PTB-3 available at http://psychtoolbox.org/, Brainard, 1997) was used for the presentation of the stimuli and for controlling the timing of all events. Experiment codes were sent from the PsychToolbox (PTB) computer to the EEG recording system via a National Instruments Card (NI-6218). Although in the still images paradigm an experiment code was sent every time a selection was presented (feedback onset time), to get more reliable feedback onset times, a photodiode was placed on the screen presenting the images. This photodiode was activated every time the feedback was presented to the subject (by whitening a small circle in the screen, where the photodiode was placed). The signal from the photodiode was digitized using a StimTracker (Cedrus Corporation) and was sent directly to the EEG system via the NI card.

2.2.2 EEG recording: EEG data and experiment codes were recorded during the entire session. An EEG cap with 48 passive electrodes, located according to the extended 10/20 international system sampled data at 256 Hz, using three synchronized g.USBamps (Guger Technologies 2016). The ground electrode was placed at the center of the forehead (at Fpz) and the reference electrode clipped to the right earlobe. Based on previously studied ErrP characteristics, this configuration ensured recording of the dominant traits of the ErrP signals and facilitated higher spatial resolution. It also increased the likelihood of finding possible sources of noise affecting the EEG signals. A dedicated computer used Matlab (v.7.11.0.584, R2010b) and Simulink (v7.6, R2010b) to record the data. In this paradigm all the sessions were collected for offline analysis and training (open-loop); no closed-loop decoding was implemented.

2.2.3 Binary-choice images task. During a session, a subject wearing an EEG cap (and seated approximately 50 cm from the PTB computer) evaluated whether Baxter's object selection was correct or not based on still images presented on a screen (while EEG data was recorded). During each trial (Figure III.1), the subject was asked to fixate on the center of an image showing two blue cups (for 1 second). This was followed by one cup's light-emitting diodes (LED) turning on, cuing to the subject the correct target. After 1.5 seconds, an image of Baxter reaching to one of the cups was presented (for 3 seconds), finishing the trial and starting a new one. The target selected by Baxter was randomly selected (with a 50% chance of selecting the incorrect target). All sessions were recorded at the MIT Distributed Robotics Laboratory, next to the Baxter robot, in order to keep a similar environment (sources of noise and distraction) as the one of the binary-choice reaching task.

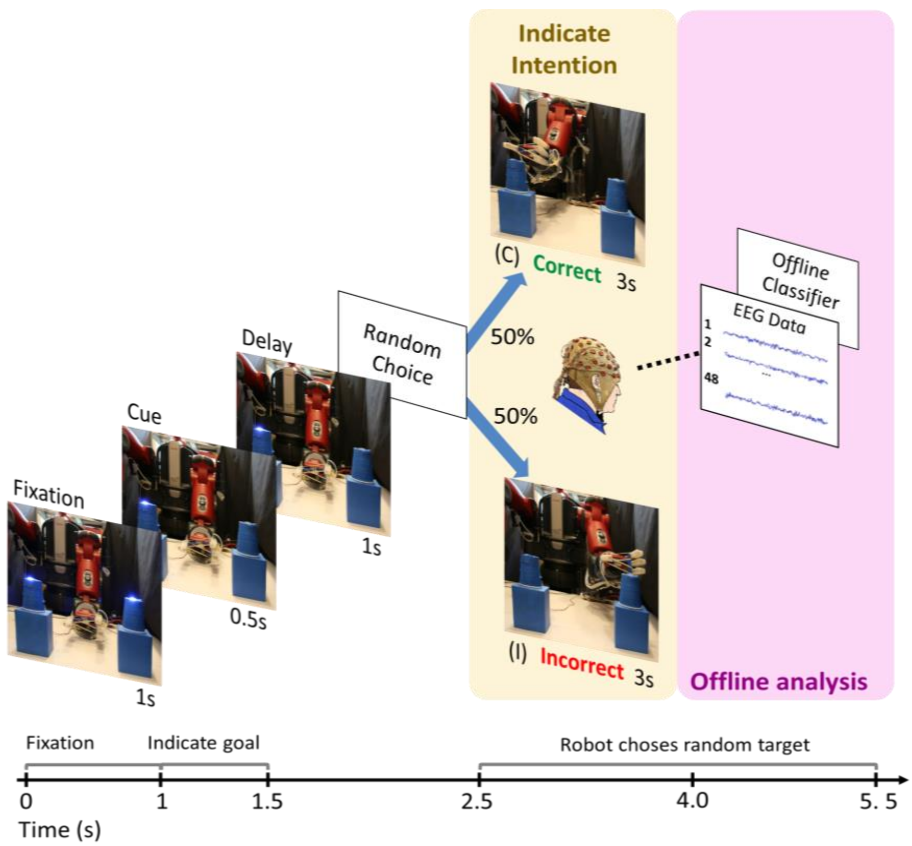


Figure III.1. Binary-choice images paradigm.

During each trial, the subject was asked to fixate on the center of an image (1 second fixation), then one of the two cup's LEDs turned on to cue the correct target (indicate goal, for 1.5 seconds). Later, an image of Baxter reaching to one of the cups was presented (indicate intention, for 3 seconds). The target selected was randomly chosen (with a 50%/50% probability of selecting the correct/incorrect target).

2.3 Binary-choice reaching paradigm

The paradigm and feedback system were designed and implemented in different platforms and programing languages. The control and classification system was divided into four major subsystems, which interacted with each other (Figure III.2): Baxter Robot, EEG System, EEG/Robot Interface, and Experiment Controller. Overall, a subject was in charge of observing the task performed by the Baxter Robot while an EEG system collected, processed and classified (on a single-trial basis) the subject's brain activity. All the processes were coordinated and timed by the Experiment Controller, which interacted with the Baxter Robot and the EEG System via the EEG/Robot Interface.

2.3.1 Experiment Controller. The experiment controller, written in Python, oversaw the experiment and implemented the chosen paradigm. It decided which target was correct for each trial, when the LEDs were turned on and off, where Baxter should reach, and generally enforced the timing of all experimental events. The Baxter robot communicated directly with the experiment controller via the Robot Operating System (ROS). The controller provided joint trajectories to Baxter's left 7 degree of freedom arm in order to indicate an object choice to the human observer and to complete a reaching motion towards that object or the opposite one. The completion or change in trajectory of the reaching motion depended on whether an ErrP was detected after the initial choice (only applicable to the closed-loop sessions). The controller also projected images onto Baxter's screen, normally showing a neutral face but, upon the detection of an ErrP by the EEG system, switching to an embarrassed face (see Figure III.3). Two indicator LEDs, used to cue the correct target, were also controlled by this subsystem via the EEG/Robot Interface.

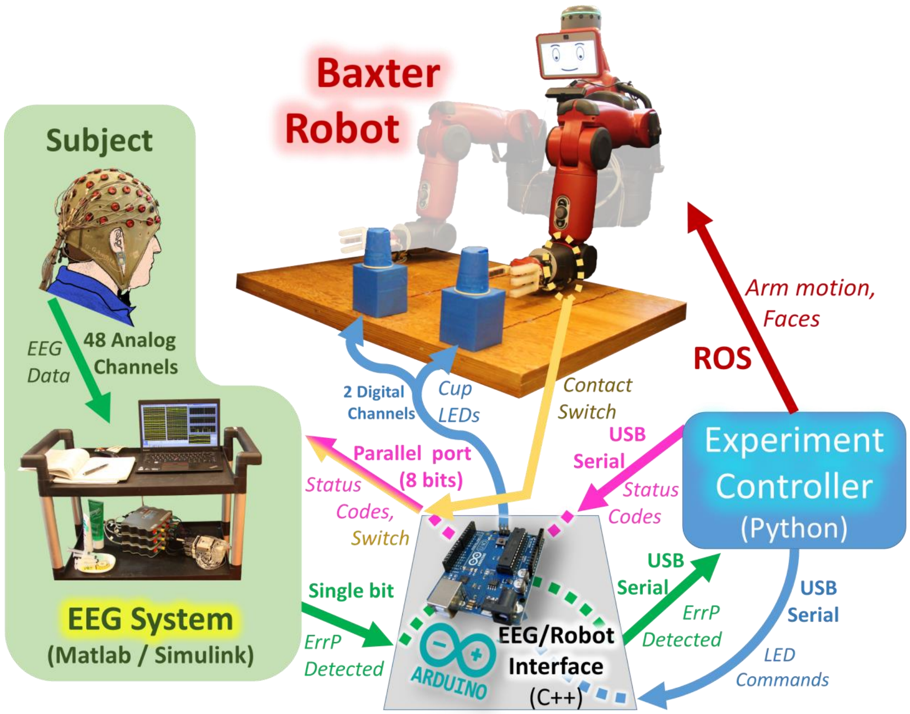


Figure III.2. Different blocks in the binary-choice reaching experiment.

Experiment setup for the direct control of a binary-choice task using ErrP signals. A subject was in charge of observing the binary-choice reaching task performed by the Baxter Robot while an EEG system (Matlab and Simulink) collected, processed and classified the EEG signals. All the processes were coordinated by an Experiment Controller (coded in Python) that controlled the indicator LEDs (activated using the EEG/Robot Interface) and interacted with the Baxter Robot (via robot operating system, ROS) and the EEG System (via the EEG/Robot Interface). To perform real-time classification of ErrP accurately, a contact pushbutton switch was used to signal feedback onset time to both the Experiment Controller and the EEG system.

The Experiment Controller was in charge of sending paradigm codes to the EEG system via the Arduino EEG/Robot Interface. The codes described events such as the trial start, target cued (LED stimulus onset), target chosen by Baxter, and robot motion. It was also in charge of receiving the command to change Baxter’s trajectory when an ErrP was detected by the EEG system. The Experiment Controller “listened” for this error signal throughout the entirety of Baxter’s reaching motion.

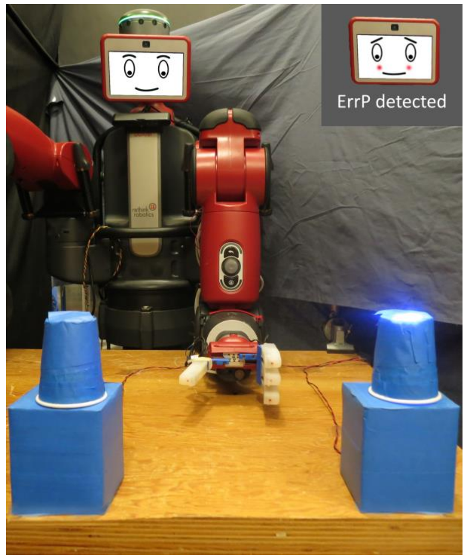


Figure III.3. Point of view of the user during the binary-choice reaching paradigm.

Indicators LEDs and the Baxter robot's left arm are centered in the user's visual field. From the subject, the LEDs and Baxter's arm are 50cm and 1m away, respectively. In this image, the right LED is cueing the correct target location. A neutral face was always displayed on Baxter's screen unless an ErrP was detected, which changed the displayed face to an embarrassed one (insert on the top-right of the picture).

2.3.2 EEG System: the EEG system was in charge of recording the EEG and behavioral data, pre-processing and decoding the EEG signals (detecting ErrP), and communicating with the EEG/Robot Interface. During each session, real-time EEG signals from a human operator were collected via 48 passive electrodes at 256 Hz, using three g.USBamps (with the ground electrode at Fpz and the reference on the right earlobe). A dedicated computer running Matlab (v.7.11.0.584, R2010b) and Simulink (v7.6, R2010b) was used to capture, pre-process, and classify (only during the closed-loop sessions) the data.

The EEG system had an 8-bit parallel port. The 7 less significant bits were inputs dedicated to collect the experiment codes and the 8th bit (only active in the closed-loop sessions) was used to output the classifier's decision to the Experiment Controller (0 when no ErrP was detected, 1 when an ErrP was found).

2.3.3 EEG -Robot Interface. The EEG/Robot interface was comprised of an Arduino Uno that controlled the indicator LEDs, forwarded messages from the Experiment Controller to the EEG system and from the EEG system to the Experimenter Controller. It sent status codes to the EEG system using 7 pins of an 8-bit parallel port connected to extra amplifier channels of the EEG acquisition system. A pushbutton switch, discussed in more detail below, was also connected directly to the 8th bit of the parallel port to inform the EEG system of robot motion. The EEG system used a single 9th bit (in the Arduino port) to send ErrP detections to the Arduino. The Arduino communicated with the Experiment Controller via USB serial.

Experiment codes that described the trial events were received by the Arduino, which forwarded them (as 7-bit codes) to the EEG system by setting the appropriate bits of the parallel port. All bits of the port were set simultaneously using low-level Arduino port manipulation to avoid synchronization issues during setting and reading the pins. Codes were held on the port for 50ms before the lines were reset. The EEG system thus learned about the experiment status and timing via these codes, and used this information for classification. In turn, the EEG system output a single bit to the Arduino to indicate whether an ErrP was detected. This bit triggered an interrupt on the Arduino, which then informed the Experiment Controller to correct Baxter's trajectory.

2.3.4 Pushbutton switch. Knowing the exact moment of arm motion initiation, i.e. the feedback onset time, was vital for reliable EEG classification. This was achieved via a hardware switch mounted to the bottom of Baxter's arm (Figure III.4). The arm's neutral position was chosen such that the switch contacted the table, but it was immediately released when the motion began. One end of the switch was wired to 5V, while the other end of the switch was passively pulled low by a resistor and wired directly to the 8th pin of the parallel port of the Arduino. This allowed the EEG system to detect the instant at which the arm motion began, much more precisely and reliably than if a software trigger was used via the Arduino parallel port (several tests demonstrated delays up to 500ms due to internal processing in the computer controlling the ROS and the Experiment Controller).

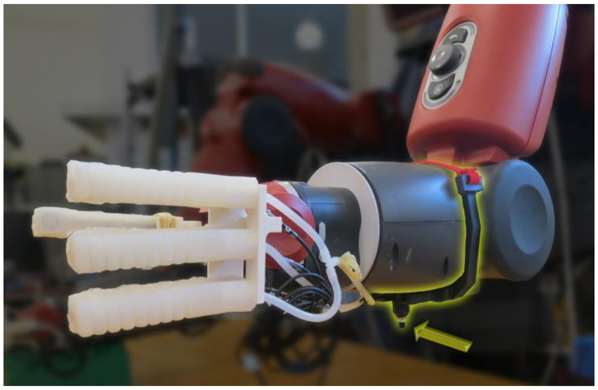


Figure III.4. Pushbutton switch used to signal the feedback onset time.

A pushbutton was installed at the bottom of Baxter's left arm (support highlighted in yellow) so upon starting a reaching movement a high signal was sent to the EEG system and the Experiment Controller to set the feedback onset time. This switch allowed precise and reliable timing of the beginning of the arm's movement.

2.3.5 Binary-choice reaching task. For decoding (on a single-trial basis) volitional commands using ErrP signals, a binary-choice paradigm (in the form of robot arm reaches) was designed and implemented (Figure III.5). A human observed whether a robot performing a reaching task made a correct or incorrect decision, and resulting EEG signals were used to influence the robot's behavior in real time (for each trial). Two versions of the same paradigm were implemented:

Open-loop sessions: No online, single-trial, classification was run while Baxter performed the target selection, and a one-stage reaching movement was used. The subjects evaluated Baxter's performance, and were aware that their EEG signals were not controlling it. Data from eleven sessions was collected. In seven of these sessions the probability of Baxter choosing the correct target was 50%, and in the remaining sessions this probability was 70%. Offline analysis of these experiments confirmed the presence of the error-related potentials and optimized the parameters for the classifier used in the closed-loop sessions. These sessions comprised five blocks of 50 trials each.

Online closed-loop sessions: The subject's EEG signals were used to control Baxter's behavior in real time, on a single-trial basis. A full session was comprised of four blocks (of 50 trials each). The first block was used to collect data for training and no classification was performed, but Baxter's controller randomly decided whether to inform Baxter of an ErrP (internally simulating the EEG classification with a 50% probability of finding an ErrP) in order to trigger a change in trajectory to the other LED (and in some cases producing an interaction ErrP if the change in trajectory was incorrect). At the end of each block, a new classifier was trained with the data collected from the current subject. Online classification was performed and used for closed-loop feedback in blocks 2-4 of these closed-loop sessions. In these sessions the probability of Baxter initially choosing an incorrect target was 30%. A total of ten sessions were recorded.

Including EEG cap preparation, a typical recording session lasted approximately 1.5 hours (taking 9 minutes each block). The paradigm was implemented at the MIT Distributed Robotics Laboratory, where the robot was located.

During a session, a subject wearing an EEG cap (seated 50 cm from the robot) judged if Baxter's object selection was correct or not while EEG data was collected (and in the closed-loop sessions, decoded). During each trial, the subject was asked to gaze at a fixation point placed directly below Baxter's arm, and in the center of two blinking LEDs, in order to avoid saccading towards the LEDs or Baxters' arm. Baxter' arm was placed in between both LEDs and at 1m from the subject (Figure III.3). At the start of a trial, both LEDs blinked (for 500ms) to cue the subject that a new trial was starting, then one of the two LEDs signaled the desired target (left or right) for 500ms. After a delay period of 750ms, Baxter reached towards one of the two targets. In the open-loop sessions Baxter made a direct reach to one of the two LEDs (chosen randomly with 50/50 and 70/30 bias for correctly selecting the target in the first 8 and in the last 3 open-loop sessions, respectively) then returned to the center of the table, waiting for the start of a new trial. In the closed-loop sessions, Baxter performed a two-stage movement towards one of the targets, which was randomly selected with a 70% bias towards choosing correctly. In the first stage the arm performed a lateral motion that informed the subject of Baxter's intended choice (releasing the pushbutton switch to initiate the EEG system's classification, Figure III.4). The second reaching stage relied on the online classifier instructing Baxter to change its trajectory or stay in the selected one conditional on detecting the presence of an ErrP.

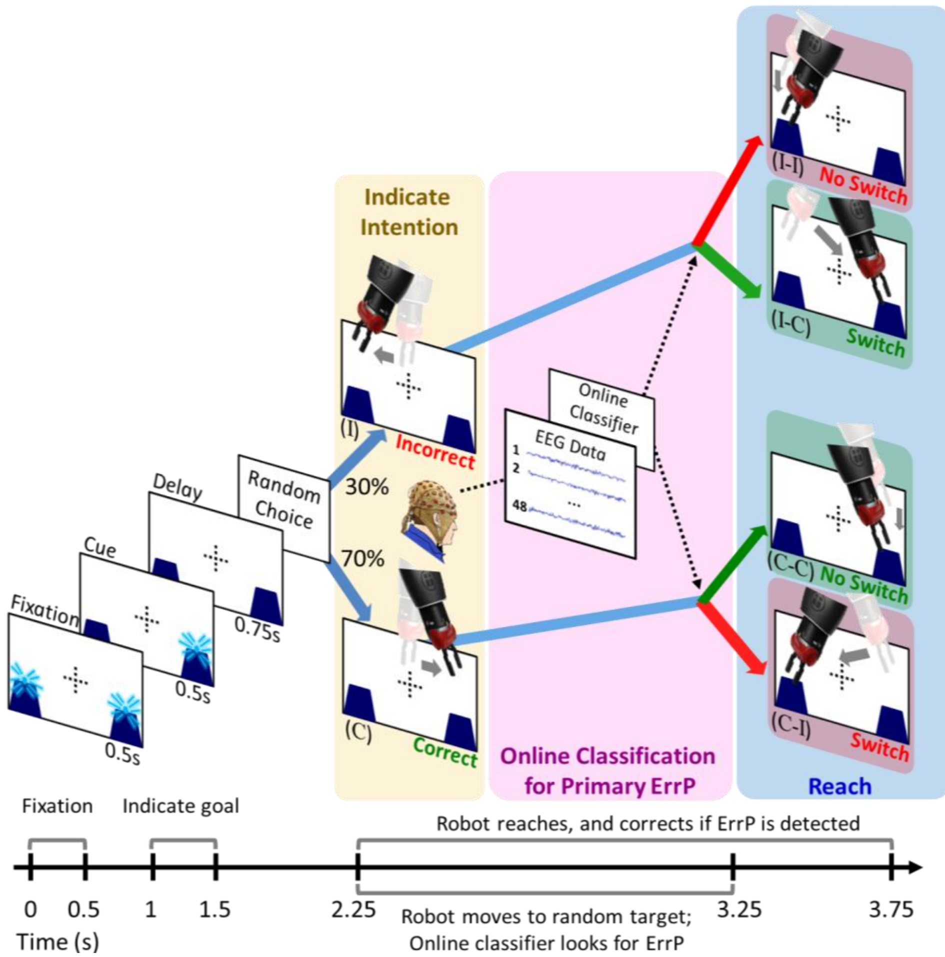


Figure III.5. Binary-choice reaching paradigm.

During a trial, both LEDs blinked to cue the start of the trial then one of them signaled the correct target. After a delay period, Baxter reached towards one of the two targets. In the open-loop it moved directly to one of the LEDs and finalizing the trial. In the closed-loop sessions, Baxter performed a two-stage movement, which was randomly selected with a 70/30 bias towards choosing correctly. In the first stage the arm performed a lateral motion that informed the subject of Baxter's intended choice. The second reaching stage relied on the detection or not of an ErrP. If no ErrP was found, the arm did not switch its trajectory, otherwise it would.

For the closed-loop sessions, if a misclassification occurred, an interaction ErrP (iErrP) should be produced. This iErrP was later analyzed and could be used in the future to include a secondary ErrP classification pipeline to further improve the decoder performance. In all the reaching task recordings, the subjects wore ear plugs and were surrounded by curtains to block any auditory and visual distractions. Baxter was located in a hallway with heavy traffic, and next to an access door that was used frequently. These distractions were an extra source of noise in our analysis since loud sounds, or the noise of people chatting, were common.

In the Results Section closed-loop online and offline results are presented. The former were obtained during the online single-trial closed-loop classification and the outcome of the ErrP classification was presented to the users (closed-loop feedback). The offline results were obtained after analyzing the closed-loop data offline (hence neither allowing feedback nor possibilities of triggering iErrP).

2.4 Offline analysis

Extensive offline analyses were performed on the open-loop data to describe the ErrP signals, possible sources of artifacts, and to set the best parameters, features, and decoder to be implemented in the closed-loop sessions. Four-fold and ten-fold cross-validation analyses were implemented. For the data collected in the closed-loop sessions, offline analyses were also performed.

For the reaching paradigm, we employed the Elastic Net and xDAWN spatial filter following recommendations shared by Alexander Barachant in a BMI competition (info available at https://github.com/alexandrebarachant/bci-challenge-ner-2015). Unlike the BMI competition, the emphasis of our approach is on performing single-trial ErrP detection in a real-time closed-loop system.

In the closed loop sessions, two different ErrP signals were extracted. Observation ErrP were evoked when Baxter's target selection was incorrect (release of the pushbutton installed under Baxter's arm provided this feedback onset time). Interaction ErrP were evoked when the decoder's classification was incorrect (for EEG analysis the feedback onset time was provided by a software experiment code). Two different iErrP signals were obtained, *2ndCorr* and *2ndIncorr*. *iErrP 2ndCorr* were decoder errors that happened after Baxter's initial target selection was correct while *iErrP 2ndIncorr* were decoder errors that occurred after Baxter's initial selection was incorrect.

2.4.1 Temporal analysis of ErrP. EEG signals were filtered (offline) using a fourth-order Butterworth filter with a pass-band frequency of 1-10Hz. Epochs from the filtered data were extracted using the feedback onset time as criteria. An epoch spanned 200ms pre-feedback to 1300ms post-feedback onset time.

Artifact rejection: artifacts in the EEG data were removed using a semi-heuristic thresholding method that involved extensive visual inspection of the signals. The mean and standard deviation (S.D.) across trials (for each channel) was computed in the interval 0-800ms post-feedback onset (for the correct and incorrect trials independently). For each channel, the mean ± kS.D. (with k being a factor multiplying the standard deviation) were determined as the 'good' trial zone. Any trial with amplitudes outside this range was labeled as 'bad' (and removed from all the channels of the analyzed data). The value of the k factor varied with the high cutoff frequency used to filter the data. For frequencies equal to 10, 15 and 20Hz, k was 4.5; for frequencies equal to 40, 100, 200, and 256Hz, k was 5.2.

Outcome condition: based on Baxter's selection outcome or the decoding outcome, epochs were grouped into correct and incorrect trials.

True target condition: based on the cued LED (correct target), epochs were grouped into left and right macrosubsets in order to study any effect the target location could have on the data. For each left and right dataset, correct and incorrect (given by Baxter's arm reach or by the decoder's classification) trials were obtained. We focused on performing decoding of only the left or right subsets in order to evaluate the effect that eye movements may have on the quality of the data.

For each channel and condition, the mean correct and incorrect traces were plotted, as well as the ErrDiff trace (the difference between the average incorrect and the averaged correct trials). Two-sample T-tests were computed for each channel to evaluate how significant the time traces were for ErrP classification (detecting a selection or decoder error). All traces were plotted on a scalp surface (using the electrode coordinates) to present the different electrode activity at specific times employing the topoplot command from the EEGLAB toolbox for Matlab (available online from https://sccn.ucsd.edu/eeglab/, Delorme & Makeig, 2004). The same approach was taken to plot the signal variability between correct and incorrect trials on the scalp (only for significant values after a two-sample T-test).

2.4.2 Feature and parameter selection analysis. Different approaches were taken to improve the quality of the signals, to better separate the features describing correct and incorrect trials, and to improve decoder performance. Through 4-fold and 10-fold cross-validation, different parameters were iterated and their decoder performance values compared. Those parameters providing the highest performance values were used for the other analysis stages.

Re-referencing: different re-referencing approaches were evaluated in order to remove common sources of noise or to highlight local activity. The re-reference options were: no re-referencing (*none*) and removing from each channel the average from all channels (*CAR*).

Spatial Filtering: to improve the signal-to-noise (SNR) ratio of the data, different spatial filtering methods were evaluated, usually separating the data into the two outcomes or classes (correct and incorrect). All the filters, when applied, reduced the number of channels in the data. The Weighted Tikhonov Common Spatial Patterns and the Fisher Event-related Potential Spatial Filter were only tested in the still images task. The xDAWN filter was only implemented in the reaching task.

xDAWN Filter: the largest eigenvectors of the averaged trace for each class (2 classes in our case, correct and incorrect) are extracted via the xDAWN algorithm (Rivet et al., 2009; Rivet et al., 2011; Barachant, 2014). The number of filters per class was related to the number of eigenvectors used to project the data into the filtered space: our analysis ranged from 2 to 5 filters per class. Each class filter was applied to the data (reducing its number of channels to the number of filters). Then the projected data was used to directly extract features or to use the filtered average class means (correct and incorrect) to be appended to each trial (see feature selection). The reader is referred to Section 5.2.1 in the Appendix for more information on the xDAWN algorithm.

Weighted Tikhonov Common Spatial Patterns (WTCSP): this type of common spatial patterns (CSP) filter uses Tikhonov regularization and weighs each electrode contribution based on a-priori knowledge of the data (Lotte & Guan, 2011). The implementation of this algorithm was done using a WTRCSP toolbox for Matlab (kindly shared by Fabien Lotte). This spatial filter reduced the channel list from 48 to a total of N electrodes by using an nth-order WTCSP filter.

Fisher Event-related Potential Spatial Filter (REFSF): the Fisher event-related potential spatial filter (Hoffmann et al., 2006) focuses on maximizing the separation between two sets of feature vectors by taking into account the temporal variance found in event-related potentials. This characteristic differentiates REFSF from CSP and WTCSP. The algorithms to compute the RESFS were run in a REFSF toolbox in Matlab (also shared by Fabien Lotte). The number of spatial REFSF filters provided the final number of reduced channels obtained after applying the filter to the data.

No filter: finally, we also evaluated the option of not spatially filtering the data.

Electrode selection: to decrease the number of features, only the electrodes with the most significant information, and less contaminated by noise, were chosen. Three different ways of performing electrode selection were implemented.

Riemann distance: using the Riemann Geometry framework and following previous work suggesting the use of Riemann distance for decoding and electrode selection (Barachant et al., 2013; Barachant et al., 2012; Congedo et al., 2013), we extracted the 35 channels that presented larger Riemann distances[3] via an iterative process adding one by one the possible channels. Initially, the average trial for each class (correct and incorrect) was computed and dimensionally reduced by projecting it to its class-specific xDAWN filter (called the reduced average trials). Each trial was

---

[3] Python code by Alexander Barachan, available in https://github.com/alexandrebarachant/bci-challenge-ner-2015

augmented by appending to it the reduced averaged traces of both classes. Then the covariance of the augmented trials (across time) was computed (for each trial). From the covariance matrices, the Riemannian mean covariance for correct (C0) and incorrect (C1) trials was computed (using a gradient descent implementation in Matlab using code adapted from the BCI competition, available at https://www.kaggle.com/c/inria-bci-challenge). From both C0 and C1 the channels were removed one by one (columns and rows representing each channel) and their Riemann distance computed. The 35 channels giving the largest Riemann distance were kept. The list of these 35 electrodes varied across subjects. The reader can find more information on the algorithm used for Riemann distance electrode selection in Section 5.1.3 of the Appendix.

Chs2Remove: after visual inspection of the EEG traces, it was clear that some channels presented more artifacts. These were usually placed near the eyes, the masseter and temporal muscles, and in the occipital area (some subjects leaned against a wall pressing the electrodes and adding artifacts). The Chs2Remove list of electrodes were all of those except T7, F7, AF7, AFz, AF8, F8, T8, P7, P8, PO3, POz, PO4, PO7, PO8, O1, Oz, and O2.

Central9List: the ErrP signal has been found to be stronger in amplitude, and more informative, in the Cz and FCz electrodes (Falkenstein et al., 1991; Gehring et al., 1993), especially for ErrP decoding (Chavarriaga & Millan, 2010). For this reason we also evaluated using only the 9 most central electrodes: FC1, FCz, FC2, CP1, C1, Cz, C2, CPz, Pz.

Type of Features: different features can be extracted from the same dataset. We explored the following options for feature extraction:

Cross-correlation features (corr): the average correct (μ_0) and incorrect trial (μ_1) was computed (across trials and for each electrode). For each electrode, the cross-correlation between each trial and the average trials of each class (μ_0 and μ_1) was obtained using the formula:

$$\rho_{i,j} = \frac{E\left[(X_i - \overline{X_\iota})(\mu_j - \overline{\mu_J})\right]}{(\sigma_x \, \sigma_\mu)}$$

where X_i is a single trial, $\overline{X_\iota}$ is the average of X_i, μ_j is the average of each class (all the correct or incorrect trials, with $j=0$ for correct, $j=1$ for incorrect), $\overline{\mu_J}$ is the average of μ_j; and σ_x and $\sigma_\mu$is the standard deviation of X_i and μ_j, respectively. All computations were done using zero lag. Finally, a cross-correlation index r was obtained by finding the difference between the average incorrect trial cross-correlation and the average correct trial cross-correlation. This analysis was performed for each electrode. For i = 0, 1 (0 for average correct trial, 1 for average incorrect) the cross-correlation index r is defined as:

$$r_i = \rho_{i,1} - \rho_{i,0}$$

Covariance features (onlycov): from each trial (and channel or reduced channel) the covariance was computed and vectorized using the tangent space mapping. These features were only used with data from the still images task.

Covariance xDAWN features (cov): the average of each class (for each of the channels, across trials) was computed. An nth-order xDAWN filter (between 2 and 5) was applied to the average correct and incorrect traces (Rivet et al., 2009). The class average traces were filtered using their respective xDAWN filter. The reduced average traces from both classes were appended to each trial (regardless of its class). From these augmented trials their covariance was computed (across the time dimension) to obtain a [*Nred* x *Nred* x *Ntrials*] feature matrix (with *Nred* as the total of reduced channels, and *Ntrials* as the total of trials). Each trial covariance matrix was vectorized by projecting it to the Riemann Geometry Space using tangent space mapping. This latter computation provided a feature vector for each trial. The reader can find in Section 5.2.2 of the Appendix all the steps for extracting the covariance xDAWN features.

Correlation of xDAWNed trial (xcorr): from the average traces of each class an nth-order xDAWN filter was computed (corrxDAWN and incorrxDAWN). Both filters were applied to each trial and then appended to create an augmented trial (with 2*nth channels). From these augmented xDAWNed trials the average trial of each class (for correct and incorrect trials) was computed (μ_0 and μ_1) and the cross-correlation between each augmented trial and the average trial (for each condition) was calculated. Later the differences of these per-trial cross-correlation values were obtained to get a cross-correlation index (for each trial).

The *corr, cov* and *xcorr* features were also augmented by performing combinations of these three. These included the *corrcov* (cross-correlation of trials and covariance of xDAWN trials), *xcorrcov* (cross-correlation of xDAWNed trials and covariance of xDAWN trials), and *corrcovxcorr* (cross-correlation of trials, covariance of xDAWN trials, and cross-correlation of xDAWNed trials).

Signal and feature scaling: different scaling and signal transformation functions were applied to the features or the signals in order to improve some characteristics in their distribution. These functions were:

Z-score: each feature was z-scored. Each feature was transformed by first subtracting its mean and the dividing it by the feature's standard deviation.

Trial z-score: each trial (and channel) was z-scored (across samples, not features).

L1-norm: each feature was normalized to have unit L1 norm.

Trial L1-norm: each trial and channel was normalized to have unit L1-norm (across the time samples).

Scale: each feature (or time sample) was linearly transformed to the range [0,1].

Trial scale: each trial was scaled by its minimum and maximum value.

None: no scaling or transformation was performed.

Frequency band: the epoched data was filtered using different pass-bands. These were 1-10Hz, 1-40Hz, 1-55Hz, and 1-80Hz. These different frequencies were only evaluated for the reaching task data.

Type of Decoder: different decoders were implemented and tested. Riemann Minimum Distance to Mean decoder (MDM), Support Vector Machine (SVM) and Linear Regression using Elastic Net shrinkage. The Riemann distance and the SVM decoders were only implemented with the images task data. Linear Regression using Elastic Net shrinkage was used with the reaching task data.

Minimum Distance to Riemann Mean decoder (MDM): This decoder used the Riemann distance computations to classify if a trial belonged to the correct or incorrect class using the minimum distance to mean criteria (Congedo et al., 2013). For each trial the covariance matrix was computed (*Ci*). Using the covariance matrices of all the trials, the Riemann Geometry mean covariance matrix was computed for each class (*C0* and *C1*, for correct and incorrect trials, respectively). The similarity of the covariance matrix of *Ci* was evaluated by computing the Riemann distance between *Ci* and *Ck* (with $k = 0, 1$). The smaller distance between *Ci* and a mean class covariance determined whether the trial belonged to that class. Due to computational restrictions during the tangent space mapping of the Riemann Geometry mean covariance matrix, covariance matrices of up to 9 reduced channels were used (the inversion of covariance matrices larger affected the stability of the computations and lead to ill-posed matrices). The 9 channels were obtained using the REFSF and WTCSP filters, or using the FC1, FCz, FC2, C1, Cz, C2, CP1, CP2 channels (these were selected because the ErrP signals have been found to present larger amplitude in these electrodes, Chavarriaga et al., 2014). A detailed presentation of the Minimum Distance to Riemann Mean algorithm is presented in Section 5.1.2 of the Appendix.

Support Vector Machine (SVM): A support vector machine decoder was implemented using the libSVM toolbox in Matlab (available from CC Chang at http://www.csie.ntu.edu.tw/~cjlin/libsvm/). A radial basis function kernel was used with parameters C and γ computed via grid-search.

Linear Regression with Elastic Net Shrinkage: initially linear regression was computed using the feature vectors and labels. The linear regression weights, the true and expected values were input into an Elastic Net. The weight of Lasso (L1) and ridge (L2) regularization was set to 0.5 (α) in most of the cases, although grid-search was also implemented to assess the optimal value (only performed on some data before setting it to 0.5). The value of λ was computed via iterative optimization performed in Matlab (lasso command) using the minimum square error (MSE) as a cost function. The Elastic Net solves the following regularization problem: Let N be the number of trials, β the linear regression weights, and X the feature matrix.

$$\min_{\beta_0,\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}(y_i - \beta_0 - x_i^T\beta)^2 + \lambda P_{0.5}(\beta)\right)$$

$$P_{0.5}(\beta) = 0.25\|\beta\|_2^2 + 0.5\|\beta\|_1 = \sum_{j=1}^{P}(0.25\beta_j^2 + 0.5|\beta_j|)$$

with the Elastic Net a new set of weights was defined. These linear regression coefficients were saved and used by the decoder in the testing trials of the closed-loop sessions.

2.4.3 K-mean clustering. Although our paradigms involved binary choices, hence two classes for classification, we explored the possibility of clustering each class into sub-clusters. The aim of this procedure was to improve decoder performance by better characterizing the data or feature space. K-means is an unsupervised classification algorithm that focuses on iteratively creating boundaries that partition the observations into clusters by associating them to the nearest mean (created by minimizing the within-cluster sum of squares). This mean serves as a prototype of the specific cluster (Tan et al., 2005). The trials from each class (correct and incorrect) were clustered using the kmean command from Matlab. For each cluster a new class label was provided so that all the clustered data could be used for classification using an augmented label set. For example: if the correct and incorrect trials were reorganized in two and three clusters, respectively, the labels of the correct trials were set to 1 and 2 (for trials belonging to each cluster), and the labels of the incorrect trials were labeled 3, 4, and 5 (each label corresponding to one cluster). Using these 5 classes, classification was performed and the decoding performance was computed by aggregating all the classification labels back into correct and incorrect.

2.5 Closed-loop binary-choice reaching decoding

At the end of each block of the closed-loop sessions, a new decoder was trained and then tested in the following block. This required two different pipelines of analysis, one for training (in Matlab) and one for testing (in Simulink).

2.5.1 Closed-loop ErrP decoding (training). Training the classifier was performed at the end of blocks 1-3. After each block, data were converted and epoched using the feedback onset time experiment codes (in the range 0-800ms post-feedback onset). For visualization purposes and bad trial removal, trials were filtered with a fourth-order Butterworth 1-10Hz passband zero-phase filter and grouped as correct or incorrect (based on the cued target and Baxter's selection). The average and standard deviation values for each channel were plotted and an automatic bad trial removal process was implemented, following the same process explained in Section 2.4.1 (*artifact rejection*). After obtaining the list of bad trials, new epochs from the current block were obtained by filtering the raw

EEG data with a fourth-order Butterworth 1-80Hz passband zero-phase filter. Bad trials were removed using the list of bad trials previously computed, and the remaining epochs were appended to the correct and incorrect trials of the sessions (training dataset). At the end of a block, the data available for training was incremented by appending these recently analyzed trials (at the end of the first block only 50 trials were available for classification but at the end of block 3 a total of 150 trials were available).

Offline analysis suggested the best parameters. Using the training dataset of the session, only the 9 most central electrodes were selected (FC1, FCz, FC2, C1, Cz, C2, CP1, CPz, CP2). The feature combination that provided better offline decoder performance was cross-correlation and covariance xDAWN.

Cross-correlation features (xcorr): the average correct (μ_0) and incorrect trial (μ_1) were computed (from all the session's correct and incorrect trials) and the cross-correlation between each trial and the average trials (μ_0 and μ_1) was obtained. Finally, the cross-correlation index r was obtained by finding the difference between the cross-correlation values previously computed. These analyses were performed for each electrode (providing 9 features per trial). These μ_0 and μ_1 matrices were saved to be used during the closed-loop decoding.

Covariance xDAWN features (cov): the average correct and incorrect trace (for each of the 9 channels, across trials) was computed. A fifth-order xDAWN filter was applied to each class to obtain an xDAWNed filtered average correct trace (corrxDAWN) and an xDAWNed filtered average incorrect trace (incorrxDAWN). Each training trial was augmented by appending to it both the corrxDAWN and incorrxDAWN traces. The covariance of each augmented trial was computed and vectorized using a tangent space mapping, giving a total of 190 features (Barachant et al., 2013). The corrxDAWN and incorrxDAWN matrices were saved for use during the closed-loop decoding (see Section 5.1 of the Appendix for a detailed explanation of the algorithm).

The cross-correlation and covariance xDAWN features were appended to create a 199 feature vector (per trial) that was used for training the decoder.

Decoder: linear regression was computed using all the training data. With the Elastic Net a new set of weights were defined and these linear regression coefficients were saved and used for the decoder in the testing trials of the closed-loop sessions.

Threshold: in the training phase the regression values were thresholded at different levels and the threshold that minimized the following biased cost function was selected.

$$Cost = \sqrt{0.7(1 - sensitivity)^2 + 0.3(1 - specificity)^2}$$

Sensitivity was defined as the probability of finding an ErrP when an incorrect target was selected by Baxter (true positive, TP). Specificity was defined as the probability of not finding an ErrP when Baxter selected the correct target (true negative, TN). More emphasis was placed on detecting ErrP since missing an error in a robotic task could mean breaking a process or injuring a person. The chosen threshold value was saved for use during the closed-loop decoding sessions.

2.5.2 Closed-loop ErrP decoding (testing). The robot triggered the start of the closed-loop classification pipeline by moving its arm to indicate object selection (feedback onset time). A window of EEG data was collected (800ms buffer from the feedback onset time) and passed through various pre-processing and classification stages. The result was detecting whether an ErrP signal was present, and thus whether Baxter committed an error in its initial selection. Preliminary offline analysis also indicated that a similar pipeline can be applied to interaction ErrP in order to boost performance in future work.

The classification pipeline (Figure III.6) used to analyze the data (in Simulink) had a pre-processing step where the raw signals were filtered and features were extracted. This was followed by a classification step where the trained classifier was applied to the processed EEG signals, yielding a linear regression value. This regression value was subjected to a threshold, which was also trained offline. The resulting binary decision was used to control the final reach of the Baxter robot.

Pre-Processing: after the feedback onset time, a buffer of 800ms from all 48 EEG channels was filtered using a fourth-order Butterworth zero-phase filter with a passband of 1-80Hz. The mean of all channels was then subtracted from each channel in order to remove noise common to all electrodes. Dimensionality reduction was then implemented by selecting only the 9 central electrode channels (FC1, FCz, FC2, C1, Cz, C2, CP1, CPz, CP2). Offline analysis revealed that in previous datasets these 9 channels were enough for proper decoder performance.

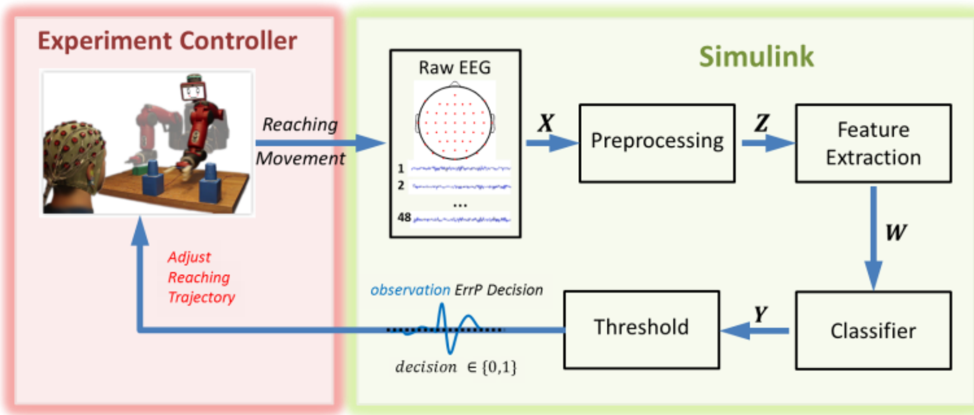


Figure III.6. Classification pipeline for the closed-loop sessions (reaching task)

Single trial classification in a closed-loop session was performed in Simulink. It involved filtering (1-80Hz) a buffer of 800ms (from feedback onset time) and extracting features (cross-correlation and covariance xDAWN features) from the filtered data. Classification was performed using a linear regression with Elastic Net shrinkage, and thresholding was done to set a decision signal. The resulting binary decision was used to control the final reach of the Baxter robot via the Experiment Controller.

Feature Extraction: features were extracted from the reduced 9 channels using two different pipelines, cross-correlation and covariance xDAWN.

Cross-correlation features: cross-correlation indexes were computed between the reduced channels and the mean correct trial and mean incorrect trial traces obtained during the training session (μ_0 and μ_1). The difference between the incorrect and correct indeces yielded a 9-features vector.

Covariance xDAWN features: each trial was augmented by appending to it the average correct-trial (across all correct training trials) and the average incorrect-trial traces obtained from training data (corrxDAWN and incorrxDAWN).The covariance of each augmented trial was computed and vectorized using a tangent space projection, giving a total of 190 features.

Both covariance and cross-correlation features were appended (a vector of 199 combined features) and used for classification.

Classifier: a linear regression decoder obtained in the training sessions (using Elastic Net regression shrinkage) was implemented. Linear regressions weights were multiplied to the feature vector to obtain a regression value.

Threshold: single-trial classification requires a binary output, i.e. the existence or the absence of an ErrP. The regression value was thresholded using the threshold value obtained in the training phase.

Decision: the binary decision was based on the result of the threshold applied to the regression value. A value of 0 indicated that no ErrP was present and the robot should complete the reach to its initial choice, while a 1 indicated that an ErrP was present and thus the robot had to switch its reach to the opposite target.

3. Results

This section explores the EEG signals obtained during both the still images and reaching tasks, and their use for ErrP detection. The EEG traces were separated into two groups based on the selection outcome: correct and incorrect. Averaged traces for each condition were computed, compared and plotted on a layout rendering the electrode location on the scalp. The ErrDiff traces (the difference between the average incorrect traces and the average correct traces, per channel) were also plotted. The results of the open-loop still images task are presented first, followed by those of the open- and closed-loop reaching task sessions.

3.1 Temporal analysis of oErrP (still images task)

The purpose of the still images task was to evaluate if ErrP signals could be triggered due to the observation of an error, and to test different ErrP classification algorithms. EEG data was grouped in correct and incorrect trials and their averages, and the ErrDiff traces were computed.

3.1.1 Observation ErrP are present in most of the EEG channels. For an example session, the average correct, incorrect and ErrDiff traces were plotted in locations resembling the electrode position on the scalp (with the nose at the top and the back of the head at the bottom, Figure III.7). The black traces (ErrDiff) present the characteristic peaks of the ErrP signal, a negative and positive peak occurring at 200ms and 350ms post-feedback onset, respectively. The amplitude of this ErrDiff trace is higher in the central electrodes, near Cz and FCz (10 μV peak-to-peak). In Figure III.7 the expanded trace belongs to Cz, and includes the standard deviations for the correct and incorrect trials (as error bars). The blue rectangles mark time samples in which the correct and incorrect trials are significantly different (two-sample T-test, $p < 0.01$). In most of the central electrodes the negative and positive peaks are also significantly different (two-sample T-test, $p < 0.01$). The electrodes in the front (first row in the plot), AF7, AFz, and AF8 show larger amplitudes due to eye movements (20 μV peak-to-peak). Although the subjects were asked to fixate to the center of the images, eye movements are still present in the traces. Most of the traces from the other open-loop still images task sessions were similar to those found in Figure III.7, except for subject 006 and 007. In these two subjects no significant differences were found between the correct and incorrect trials and the ErrDiff trace did not present the negative and positive peaks found in the other subject's traces. The average correct, incorrect and ErrDiff traces of all the other sessions for the still images task can be found in the Appendix.

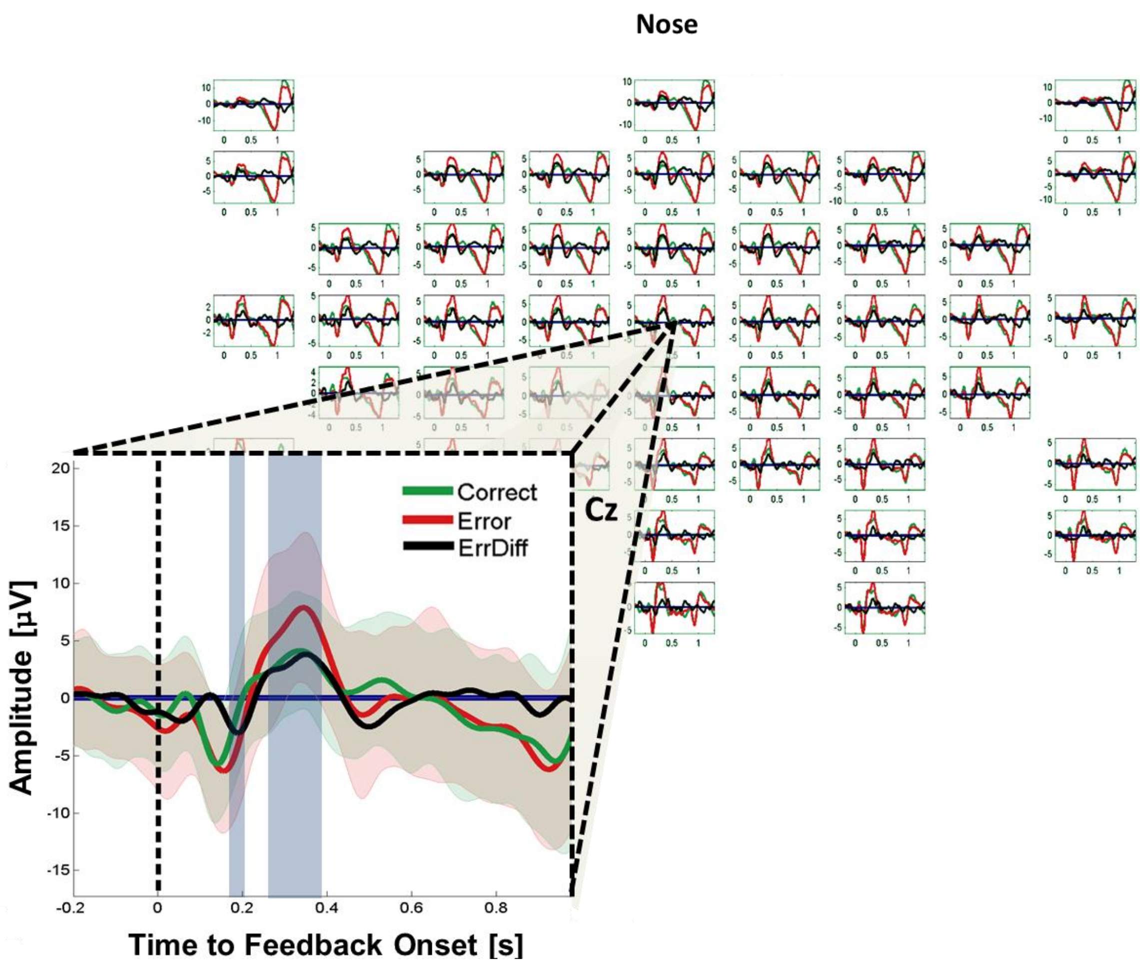


Figure III.7. Average correct, incorrect, and ErrDiff traces (still images task).

Example trial average during the still images tasks (subject 009). All electrodes positioned in a similar location to the one in the EEG cap. Top row has the front electrodes; bottom row has occipital electrodes. The traces are the average correct (green), average incorrect (red) and ErrDiff (black); with the ErrDiff trace representing the difference between the average incorrect and correct traces. Error bars depict standard deviation of correct (light green) and incorrect (light red) trials. X axis: time to feedback onset (seconds). Y axis: signal amplitude (µV). The signal expanded belongs to channel Cz. Blue rectangles mark the time samples that are significantly informative about Baxter's selection outcome (difference between correct and incorrect, two-sample T test, $p < 0.01$).

3.1.2 Components of the oErrP signal. The ErrDiff trace in all electrodes and its topographic distribution in the Cz electrode of an example session (subject 009) is plotted in Figure III.8. Almost all channels show a decrease and increase of activity at similar time points. AF7 and AF8 present larger negative deflections at the end of the trial (dark blue) probably due to eye movements. The middle plot presents the traces from the Cz electrode that was expanded in Figure III.7. In the interpolated activity (bottom plot) we found at 188ms a negative fronto-central peak that later transitioned to a centro-parietal positive peak happening 344ms post-feedback onset. This was followed by another fronto-central negative peak at 509ms.

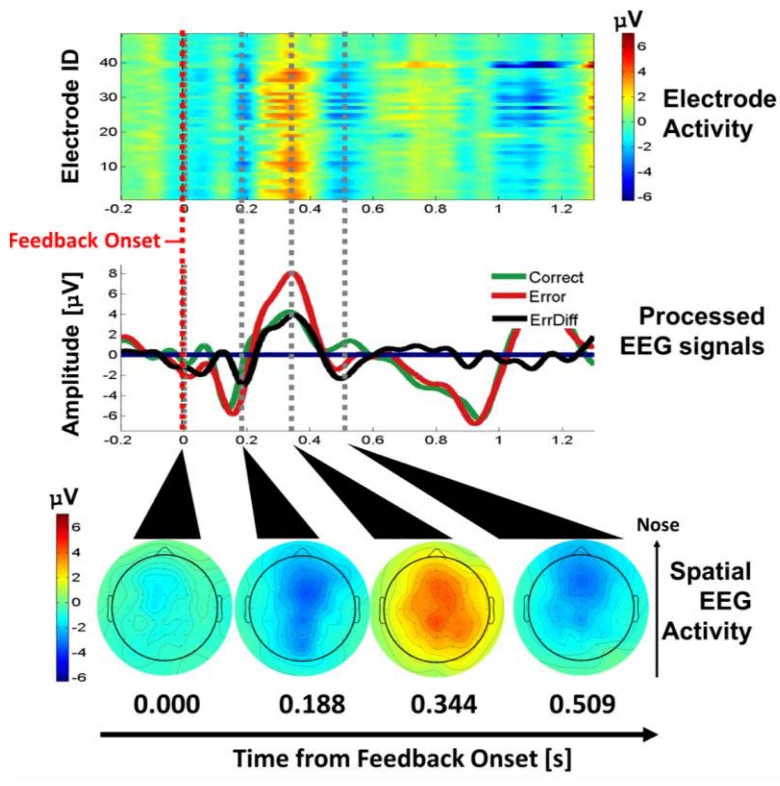


Figure III.8. ErrDiff topography for oErrP (still images task).

ErrDiff trace in all electrodes and its topographic distribution for an example session (subject 009). Top plot: ErrDiff activity for all electrodes (Y axis) for an epoch in the range -0.2s to 1.3s post-feedback onset. X axis: time to feedback onset (seconds). Cool (blue) and hot (red) colors represent negative and positive amplitude values, respectively (measured in μV). Middle plot: average correct (green), average incorrect (red), and ErrDiff (black) traces from the Cz electrode. X axis represents the time to feedback onset. Y axis depicts the signal amplitude (μV). Bottom plot: scalp distribution of the activity of all the channels for specific times (same color bar as the one in the top plot). Negative, positive, and negative peaks at 188, 344, and 509ms post-feedback onset, respectively.

3.2 Offline ErrP detection (still images task)

Four-fold cross-validation was implemented to evaluate the best parameters for offline ErrP classification. A Riemann MDM and a SVM decoder were used for classification. The classification performance using the REFSF, WTCSP spatial filters and no spatial filter were compared. All these analyses also included different scaling options: l1norm, zscore, scale, and none. Two different metrics of performance were used. Decoder performance (classification accuracy) and Area Under the Curve (AUC). Decoder performance evaluates the proportion of trials correctly classified (correct trials being classified as correct and incorrect trials being classified as incorrect). In the still images task we can use this metric (since there is a similar chance for an incorrect or correct trial to occur). If the probability of an incorrect trial is biased, decoder performance will not be an appropriate measure unless chance is specified. The AUC evaluates the Receiver Operating Characteristic (ROC) curve and takes a weighted measure of both true positives and true negatives classification values (diminishing any bias given by an unbalanced dataset).

3.2.1 Covariance features without spatial filtering provide the best oErrP classification performance values. In Figure III.9, left plot, are presented the AUC (continuous lines) and classification accuracy (dashed lines) values for the Riemann

MDM decoder for all the subjects in the still images task (similar analyses were performed with the SVM decoder). These analyses used covariance features (without any xDAWN spatial filtering or corrxDAWN and incorrxDAWN augmenting). The color traces represent different feature reduction criteria: no feature reduction (green), PCA (red), and Fisher score (blue). The dotted line represents chance. The Riemann MDM decoder was stable across feature selection criteria, although with an overall low performance. Different signal scaling and spatial filters did not improve decoder performance. The best results during offline analysis suggested that we use covariance features with no scaling. The mean decoder performance (across sessions, and using the best parameters) for WTCSP, REFSF, and none spatial filters, and for the Riemann MDM and SVM decoders are presented in the right plot of Figure III.9. The filter implementation did not provide any signal improvement nor increased decoder performance. Not using a spatial filter allowed mean performance values of 0.55 and 0.58 for the Riemann MDM and SVM decoders, respectively.

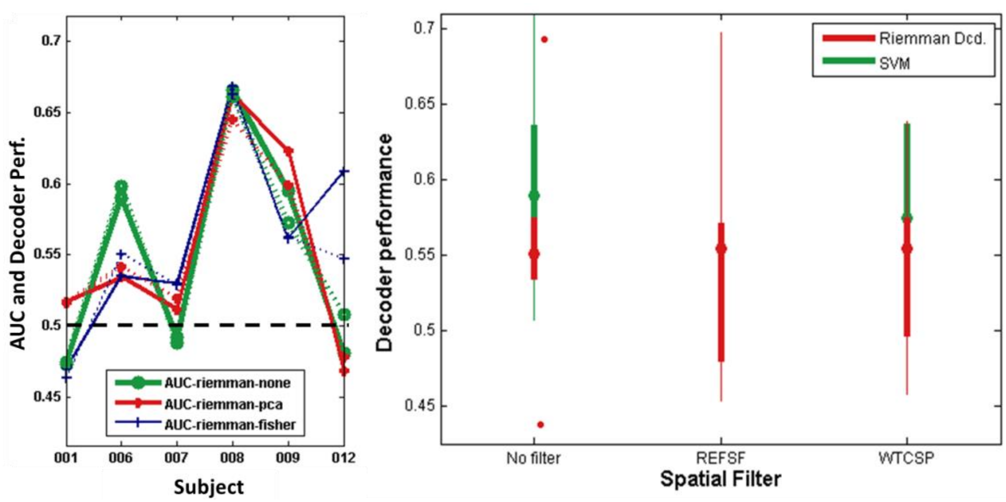


Figure III.9. AUC and decoder performance four-fold cross-validation values for all the sessions and across session performance for different spatial filters (still images task).

Left plot: AUC (continuous lines) and decoder performance (dashed lines) values for the Riemann (left plot) and SVM (right plot) decoders for all the images task subjects. Covariance features were used in these analyses. Color traces represent different feature reduction criteria: no feature reduction (green), PCA (red), and Fisher score (blue). Chance is presented as a dotted line. Right plot: mean decoder performance (across sessions, and using covariance features with no scaling) for WTCSP (right plot), REFSF (center plot), and no (left plot) spatial filters. Median and quartile distribution for Riemann MDM (red) and SVM decoders (green). The no spatial filter implementation provides the best performance values. All results obtained after four-fold cross-validation.

3.3 Temporal analysis of iErrP (reaching task)

Although the ErrP traces were found during the still images task, single-trial classification proved to be challenging. To improve decoder performance and better engage the subjects, as well as to make the task more like a real-world application, the reaching task was implemented. One important aspect was the effect the real life reaches may have on the engagement of the subjects, which could improve signal ErrP quality.

3.3.1 Observation ErrP are present in most of the EEG channels. The average activity for the correct, incorrect and ErrDiff traces (across trials) were plotted in a manner resembling the electrode layout in the scalp (nose at the top, Figure III.10). The ErrDiff traces (black) presented the negative and positive peaks of the ErrP. At 240ms and 450ms post-feedback onset we found the negative and positive peaks, respectively. The frontal and central electrodes showed the larger amplitudes (10 μV peak-to-peak). The expanded plot belongs to the Cz channels and includes the standard deviations for the correct and incorrect trials (as error bars). The blue rectangles mark the time samples that are significantly informative of the decoder outcome (two-sample T-test, $p < 0.01$). Most of the electrodes (mainly central and frontal) are significantly informative at the negative and positive peaks. The AF7 and AF8 electrodes showed larger peak-to-peak amplitudes (20 μV). These channels were not used for ErrP decoding since the large amplitudes were related to eye movements. The same traits were found in the other subjects. The average correct, incorrect and ErrDiff traces of all the other sessions for the reaching task can be found in the Appendix.

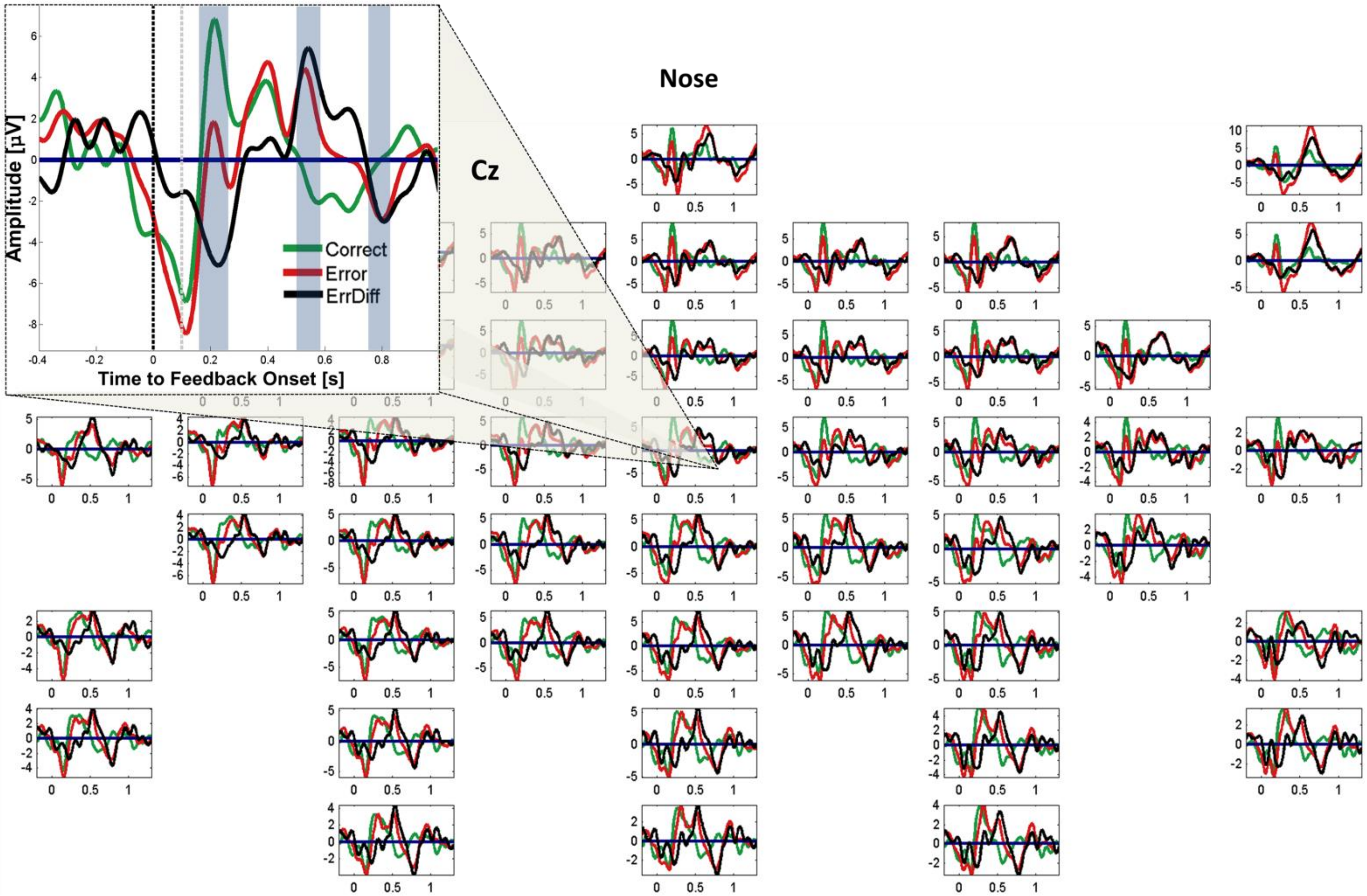


Figure III.10. Average correct, incorrect, and ErrDiff traces (reaching task oErrP).

Average trials for an example session (subject 014) of the reaching task. All electrodes positioned in a similar location to the one in the EEG cap. At the top are the frontal electrodes. The traces are the average correct (green), average incorrect (red) and ErrDiff (black) of Baxter's target selection (oErrP). ErrDiff represents the difference between the average incorrect and correct traces. X and Y axes represent the time to feedback onset (seconds) and the signal amplitude (µV), respectively. The signal expanded belongs to channel Cz. Blue rectangles mark the time samples with significant differences between correct and incorrect trials (two-sample T test, $p < 0.01$).

3.3.2 Components of the oErrP trace. After cuing the correct target, an incorrect selection made randomly by Baxter triggers an observation ErrP. Figure III.11 presents, from an example session, the scalp activity of the ErrDiff traces during observation ErrP (from subject 014)Figure III.8. All the channels present increases (red) and decreases (blue) in activity at similar time points. The middle plot presents the traces from the expanded Cz electrode. In the bottom plot, a fronto-central negative peak is found at 243ms post-feedback onset, followed by a central positive peak at 446ms, and by another negative inflexion at 822ms post-feedback onset. The timing and location of this ErrDiff is similar to the one found in the still images tasks.

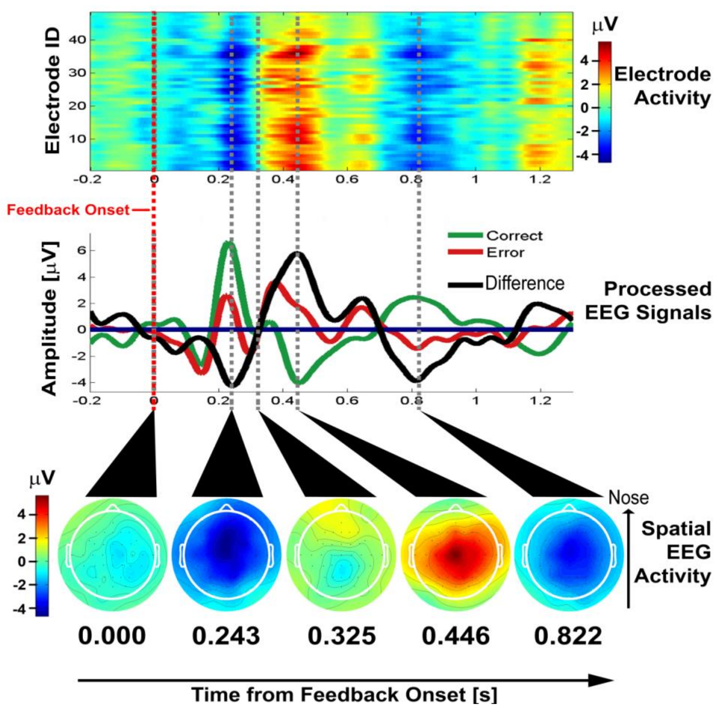


Figure III.11. ErrDiff topography for oErrP (reaching task)

Topographic distribution of the ErrDiff traces from all the electrode of an example session (subject 014). Top plot: ErrDiff activity for all electrodes (Y axis) for an epoch from -0.2s to 1.3s post-feedback onset (X axis). Cool (blue) and hot (red) colors represent negative and positive amplitude values, respectively (μV). Middle plot: average correct (green), average incorrect (red), and ErrDiff (black) traces from the Cz electrode. X axis represents the time to feedback onset. Y axis depicts the signal amplitude (μV). Bottom plot: scalp distribution of the activity of all the channels for specific times (same color bar as the one in the top plot). Negative, positive, and negative peaks are present at 243, 446, and 822ms post-feedback onset, respectively.

3.3.3 Interaction ErrP are present in most of the EEG channels. As in the oErrP case, most of the channels show the presence of iErrP. Figure III.12 presents the average activity for the correct, incorrect and ErrDiff traces (across trials) from an example subject. The traces were plotted to resemble the electrode layout in the scalp (with nose at the top). The ErrDiff traces (black) presented the negative and positive peaks of the ErrP. At 300ms and 600ms post-feedback onset we found the negative and positive peaks, respectively. The frontal and central electrodes showed the larger amplitudes (15 μV peak-to-peak). The expanded plot belongs to the Cz channels and includes the standard deviations for the correct and incorrect trials (as error bars). The blue rectangles mark the time samples that are significantly informative of the decoder outcome (two-sample T-test, $p < 0.05$). Most of the electrodes (mainly central and frontal) are significantly informative at the negative and positive peaks. The average correct, incorrect and ErrDiff traces of all the other sessions for the reaching task can be found in the Appendix.

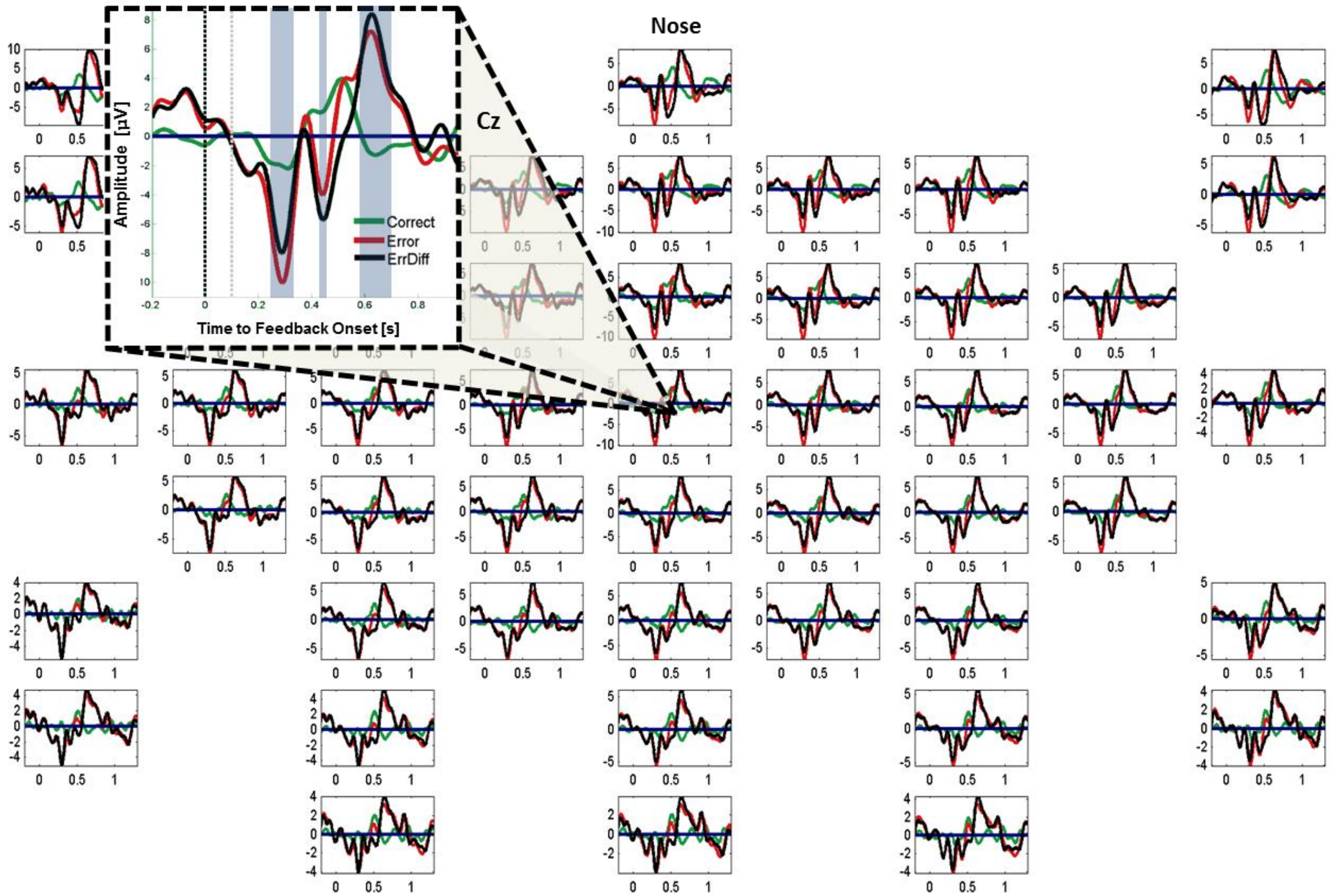


Figure III.12. Average correct, incorrect, and ErrDiff traces (reaching task iErrP 2ndCorr).

Average trials for an example session (subject 021) of the images task. All electrodes positioned in a similar location to the one in the EEG cap. At the top are the frontal electrodes. The traces are the average correct (green), average incorrect (red) and ErrDiff (black) of Baxter's target selection (oErrP). ErrDiff represents the difference between the average incorrect and correct traces. X and Y axes represent the time to feedback onset (seconds) and the signal amplitude (μV), respectively. The signal expanded belongs to channel Cz. Blue rectangles mark the time samples with significant differences between correct and incorrect trials (two-sample T test, $p < 0.05$).

3.3.4 Components of the iErrP trace. Closed-loop classification increased the subject's engagement and interest in the task. Nevertheless, even in this situation the ErrP classifier did not perform perfectly. Several classification errors occurred, triggering a secondary ErrP signal (iErrP). These traces were aligned using experiment codes (sent via the EEG/Robot Interface) related to the secondary feedback onset (when Baxter changed trajectory or stayed in the current one). Two types of iErrP occurred: when the trial was correct and Baxter changed its trajectory (2ndCorr), or when Baxter did not change its trajectory but the initial selection was incorrect (2ndIncorr). In both cases the error is a result of incorrect classification (driven by the subject's EEG activity). In Figure III.13 we present the scalp activity of the ErrDiff traces for iErrP (2ndCorr), the trials that were correctly chosen by Baxter but that the ErrP decoder labeled as incorrect (changing Baxter's trajectory). This iErrP showed similar traits and timing for negative and positive peaks in all the electrodes. A remarkable difference is the amplitude of the signal, twice the one seen in the observation ErrP (20 μV peak-to-peak, compared to 10μV in the oErrP). Negative, positive, and negative peaks were found at 442, 611, and 951ms post-feedback onset, respectively. All the peaks were located in the fronto-central region.

The timing of the iErrP peaks was slightly shifted in comparison with the oErrP ones. This is probably because the feedback onset time for the iErrP was not timed precisely using a switch or photodiode. Rather, we used an assessment of the subject's perception of the second feedback onset time and software codes. These codes were returned by the Arduino at the time when the arm was commanded to change its trajectory (when an oErrP was classified online) or at the end of the movement[4] (if no oErrP was classified online and the trial was labeled as correct). In the latter case, when the secondary feedback implied no change in trajectory, its onset time had to be estimated since no change in Baxter's behavior cued the subjects of this outcome. This situation occurred a) when Baxter's first selection was correct and the classifier did not find an oErrP and b) when the primary selection was incorrect and the classifier did not detect this error.

For the trials with no change in trajectory, we estimated the secondary feedback onset time as the difference between the end of the movement and the average time that took Baxter to wait for a classification bit (from the EEG system) and to prompt the EEG system that the trial was classified as correct. In average, 865ms took Baxter's experiment controller to process receiving a classification bit (labeling the trial as correct) and prompting the EEG system of such reception.

[4] After completing the reaching and returning to the center of the table

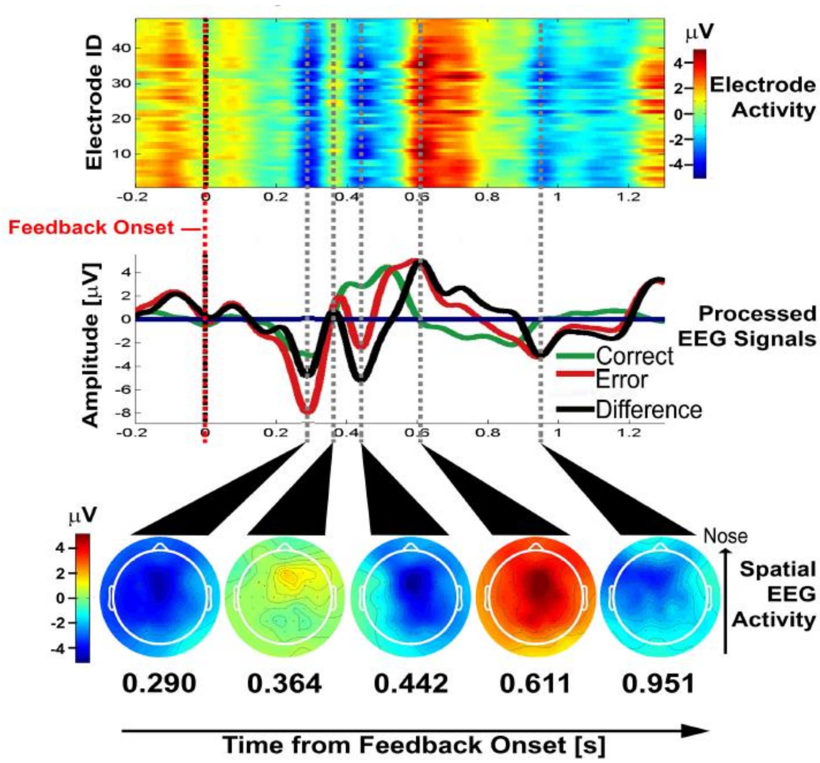


Figure III.13. Interaction ErrP topography (reaching task, 2ndCorr)

ErrDiff (after a correct Baxter choice, 2ndCorr) topographic distribution for all the electrodes of an example session (subject 021). Top plot: ErrDiff activity for all electrodes (Y axis) for an epoch from -0.2s to 1.3s post-feedback onset (X axis). Cool (blue) and hot (red) colors represent negative and positive amplitude values, respectively (μV). Middle plot: average correct (green), average incorrect (red), and ErrDiff (black) traces from the Cz electrode. X axis represents the time to feedback onset. Y axis depicts the signal amplitude (μV). Bottom plot: scalp distribution of the activity of all the channels for specific times (same color bar as the one in the top plot). Negative, positive, and negative peaks found at 442, 611, and 951ms post-feedback onset, respectively.

3.3.5 oErrP and iErrP are similar across different task and conditions. In Figure III.14 are presented the ErrDiff traces obtained under different task and conditions. Although with some variations in time, a consistent pattern is found: a negative-positive-negative-positive pattern of peaks and troughs. The still images oErrP negative and positive peaks are found earlier than those in the reaching oErrP and iErrP. Also, the reaching task oErrP peaks are found earlier than those from the iErrP. In addition to different engagement levels but also the accuracy with which the feedback onset time was established may have influenced these results.

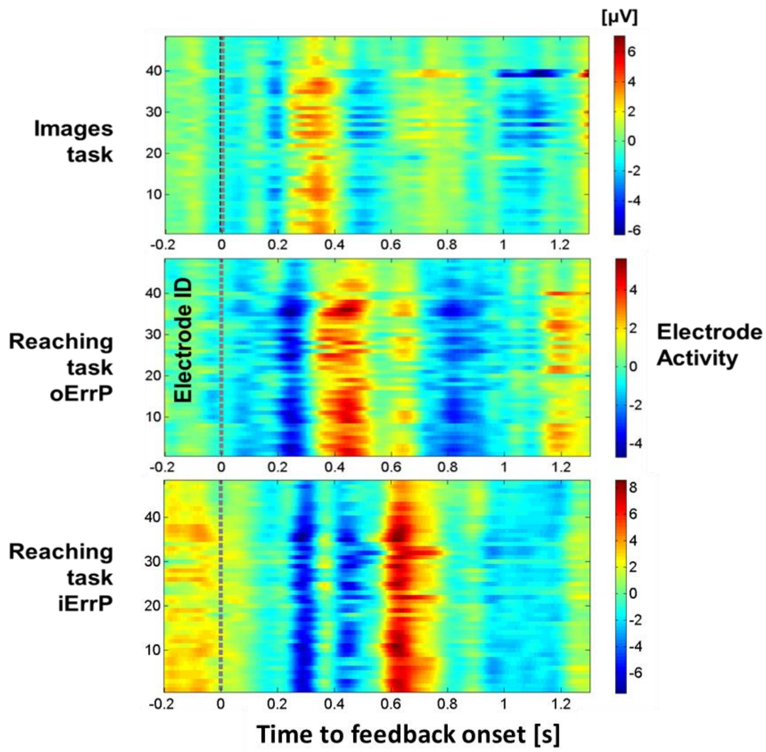


Figure III.14. Comparison of the peak times of ErrDiff for the still images oErrP, the reaching oErrP, and the reaching iErrP.

ErrDiff activity (from averaged incorrect and correct trials) for both tasks and type of ErrP. Top plot: images task ErrDiff from oErrP. Middle plot: reaching task ErrDiff from oErrP. Bottom plot: reaching task ErrDiff from iErrP (2ndCorr). X axis represents the time to feedback onset from -0.2 to 1.3s. Y axis denotes the electrode ID. Cool (blue) and hot (red) colors are negative and positive amplitude values, respectively (μV). A shift in the negative and positive peaks is present in the reaching task ErrDiff.

If we take only the naïve subjects that participated in the closed-loop sessions (removing subject M) and average the ErrDiff traces from the Cz electrode, we get a population trace for the ErrP and both iErrP (Figure III.15). Two differences between the oErrP and the iErrP (2ndCorr) ErrDiff were found: the amplitude of the average ErrDiff (across sessions) from the iErrP signals is almost double that of the average of the ErrDiff from the oErrP, and the shape of the ErrDiff from the iErrP (2ndCorr) across subjects is generally more consistent. The same cannot be stated for the iErrP (2ndIncorr), partially because the number of trials available is reduced by separating the iErrP in 2ndCorr and 2ndIncorr (see

Table III.1). These differences allow for further disambiguation of the presence of an error in the EEG signal, leading to better classification performance (see Section 3.5).

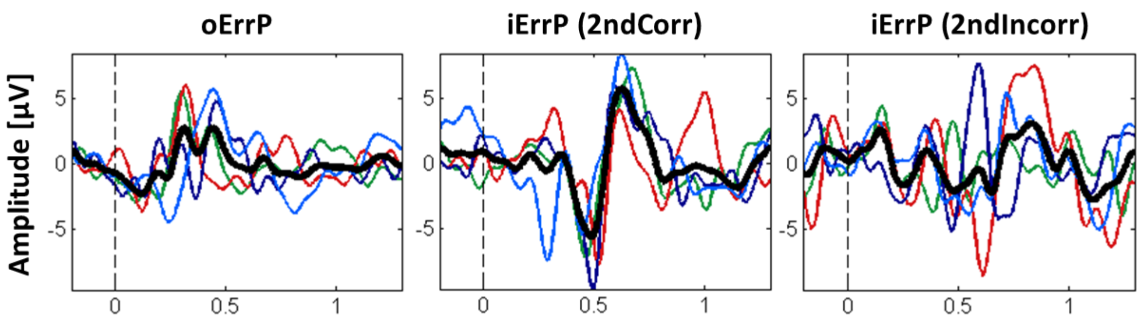


Figure III.15. Population ErrDiff traces for the oErrP and both iErrP.

ErrDiff traces for the oErrP and both iErrP (across all naïve subjects, except subject M). X axis: time to feedback onset from -0.2 to 1.3s. Y axis: signal amplitude (µV). All traces share the same scale in the Y axis. From left to right: oErrP (due to Baxter initial target selection), iErrP 2ndCorr (iErrP after a correct target selection executed by Baxter), and iErrP 2ndIncorr (iErrP after an incorrect target selection made by Baxter). Each color represents one subject. Black traces show the average trace (across subjects). The iErrP 2ndCorr signals presented larger amplitudes (almost twice that of oErrP) and provided better classification performance.

Table III.1. Number of trials for the correct and incorrect outcome of oErrP and iErrP during the closed-loop online sessions (naive subjects only).

List of the number of correct and incorrect of trials (after the bad trials were removed) for the oErrP and the iErrP scenarios (data from the naïve subjects only). Since the iErrP can be caused after a correct or incorrect Baxter choice, the number of trials for iErrP 2ndCorr and iErrP 2ndIncorr are presented separately.

Condition	Number of trials				
Subject Number	**017**	**018**	**019**	**020**	**021**
Correct (oErrP)	102	107	104	103	138
Incorrect (oErrP)	50	51	46	51	48
Correct (iErrP 2ndCorr)	74	68	55	65	111
Incorrect (iErrP 2ndCorr)	34	34	44	43	33
Correct (iErrP 2ndIncorr)	14	13	20	27	17
Incorrect (iErrP 2ndIncorr)	35	40	29	25	32

3.4 Offline ErrP detection (reaching task)

With the reaching task data, different parameters were evaluated in multiple offline simulations, running 10 iterations of 10-fold cross-validation. Different passband filters, type of features, re-referencing approaches, and electrode selection functions were evaluated. All channels were xDAWN filtered using 5 filters per class and the features were z-scored. A linear regression decoder with Elastic Net shrinkage was used for ErrP classification. We present a summary of some of the analyses across all the reaching task sessions (for the oErrP and iErrP). Similar analyses to the ones presented here were performed on the data separated on left and right subsets (when the correct target was either left or right) or regrouped using k-means clustering. None of these approaches improved decoder performance.

In the k-means clustering case the overall performance was affected negatively. This suggests that the correct and incorrect trials presented a unimodal distribution so only one cluster could be used to represent each class. Moreover, separating the trials into sub-clusters decreased the number of observations available for training.

With the left and right subgroups a different decoder for each group was tested (under different parameters) and no significant improvement in the classification performance was obtained. This indicates that for the Baxter reaching task, whether the Baxter arm moved to the right or to the left was inconsequential to the characteristic ErrP signal. We used the AUC performance metric to account for the unbalanced number of trials in some of the sessions (when the selection of a correct target was bias to 70%).

All the open-loop sessions (except the last 3) presented similar number of correct and incorrect trials (balanced) whereas the closed-loop sessions were all biased (with a 70% chance of Baxter selecting the correct target).

3.4.1 Parameter evaluation: Under different parameters, the classification AUC (average across 10 iterations of 10-fold cross-validation) of the oErrP was plotted for all open- and closed-loop sessions (Figure III.16). No parameter combination reflected a clear advantage for decoding oErrP. Nevertheless, some subjects presented better AUC values than others. In the three sessions of subject 016 the AUC was above 0.75 for most of the parameter combinations. On the other hand, subject 001 showed, most of the time, AUC values below 0.65. These differences affect the overall performance values, and demonstrate the inter-subject variability. If we set aside the open-loop sessions and focus on the closed-loop sessions (Figure III.17), we found that subject 019 (subject M) has the lowest performance values across different parameter combinations (close to chance).

For the iErrP (in the closed-loop sessions only), two different datasets were analyzed. Trials in which the correct target was initially correctly selected by Baxter (2ndCorr), and trials in which the correct target was initially incorrectly selected by Baxter (2ndIncorr). With data from each subgroup different parameters were evaluated. Figure III.18 presents the offline analysis results for the 2ndCorr subset. The overall classification performance across all sessions is better than the one obtained using the oErrP. Most of the sessions presented AUC values above 0.8, except subject 019, whose results were below 0.6, close to the values obtained in the oErrP analysis. Similar results were obtained with the 2ndIncorr subset (Figure III.19) although the AUC values were lower than the ones obtained with the 2ndCorr subset. Again, subject 019 presented poor performance across all possible parameter combinations.

From all the sessions and possible combinations (Figure III.20), the highest across subject AUC performance (0.674) for the oErrP was obtained with data band-pass filtered in the 1-10Hz range, using the Chs2Remove electrode selection function, and with the *corrcov* features. For the iErrP (2ndCorr), the highest mean AUC (0.879) was found with data filtered in the 1-10Hz range, using the *corrcov* features. The iErrP (2ndIncorr) data subset presented the highest performance values (0.782) when filtered in the 1-40Hz range, using the Chs2Remove electrode selection algorithm, and employing the *corrcovxcorr* feature combination. The Chs2Remove electrode selection function showed the least variability in the population results (more stable across subjects). Overall (for oErrP, iErrP 2ndCorr and 2ndIncorr), data filtered in the range 1-80Hz produced better performance.

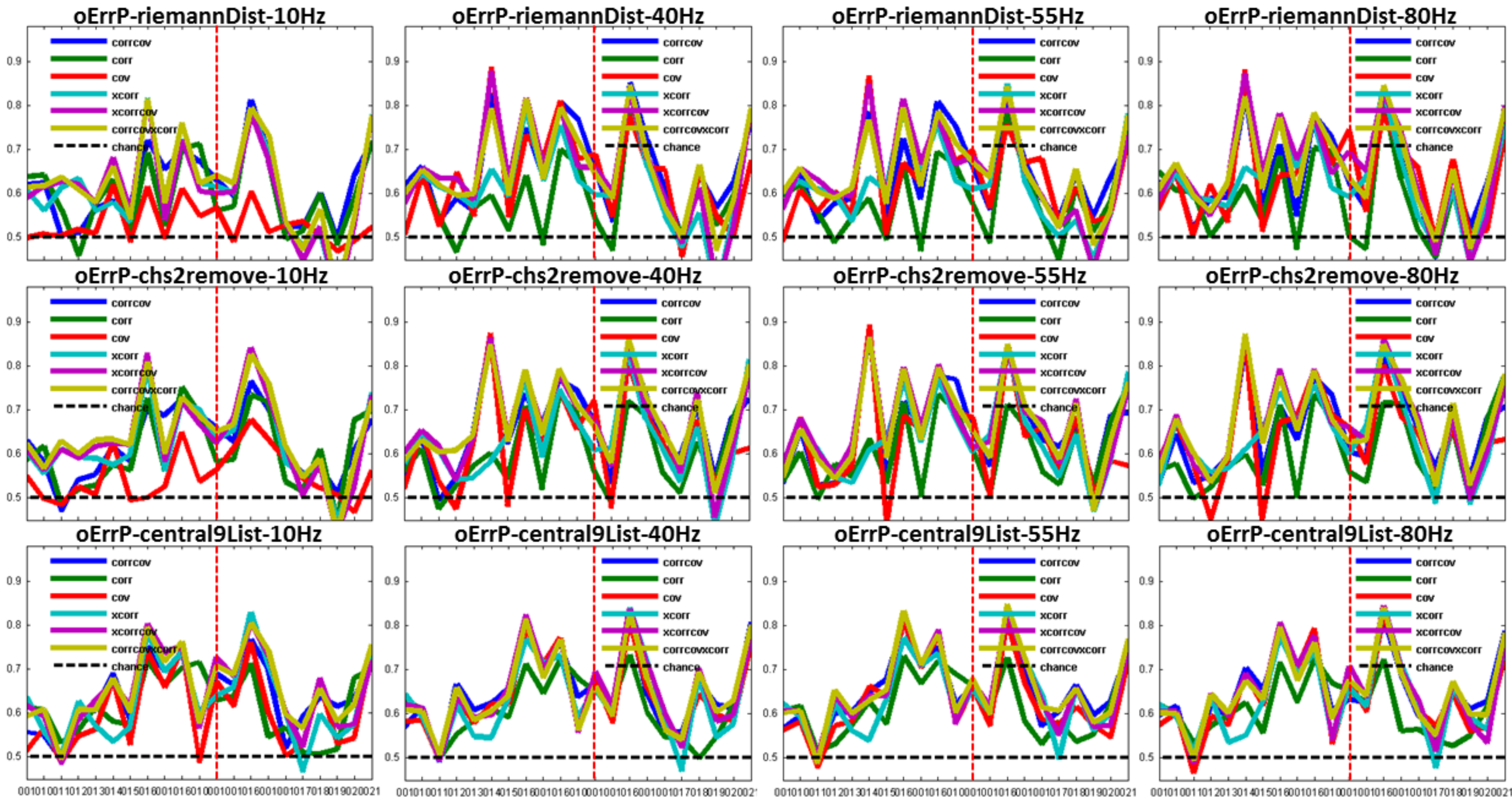


Figure III.16. AUC for different type of features, electrode selection function and pass-band frequencies for the oErrP across all open- and closed-loop sessions.

AUC for multiple parameters evaluated with 10 iterations of 10-fold cross-validation (channels reduced using a xDAWN filter of 5^{th} order and features z-scored). The Y and X axes represent the AUC and the open- (left from red dotted line) and closed-loop sessions, respectively. From left to right: columns represent 1-10, 1-40, 1-55, and 1-80Hz bandpass filter. From top to bottom: Riemann distance, Chs2Remove, and Central9List electrode selection function, respectively. Color traces are different feature extraction functions: *corrcov* (blue), *corr* (green), *cov* (red), *xcorr* (light blue), *xcorrcov* (magenta), *corrcovxcorr* (yellow). The black dashed line is chance. Each subject has similar AUC performance regardless of the parameters. Subject 019 showed low AUC performance values. The *corr* feature (green) presented the lowest performance across all subjects and passband filter.

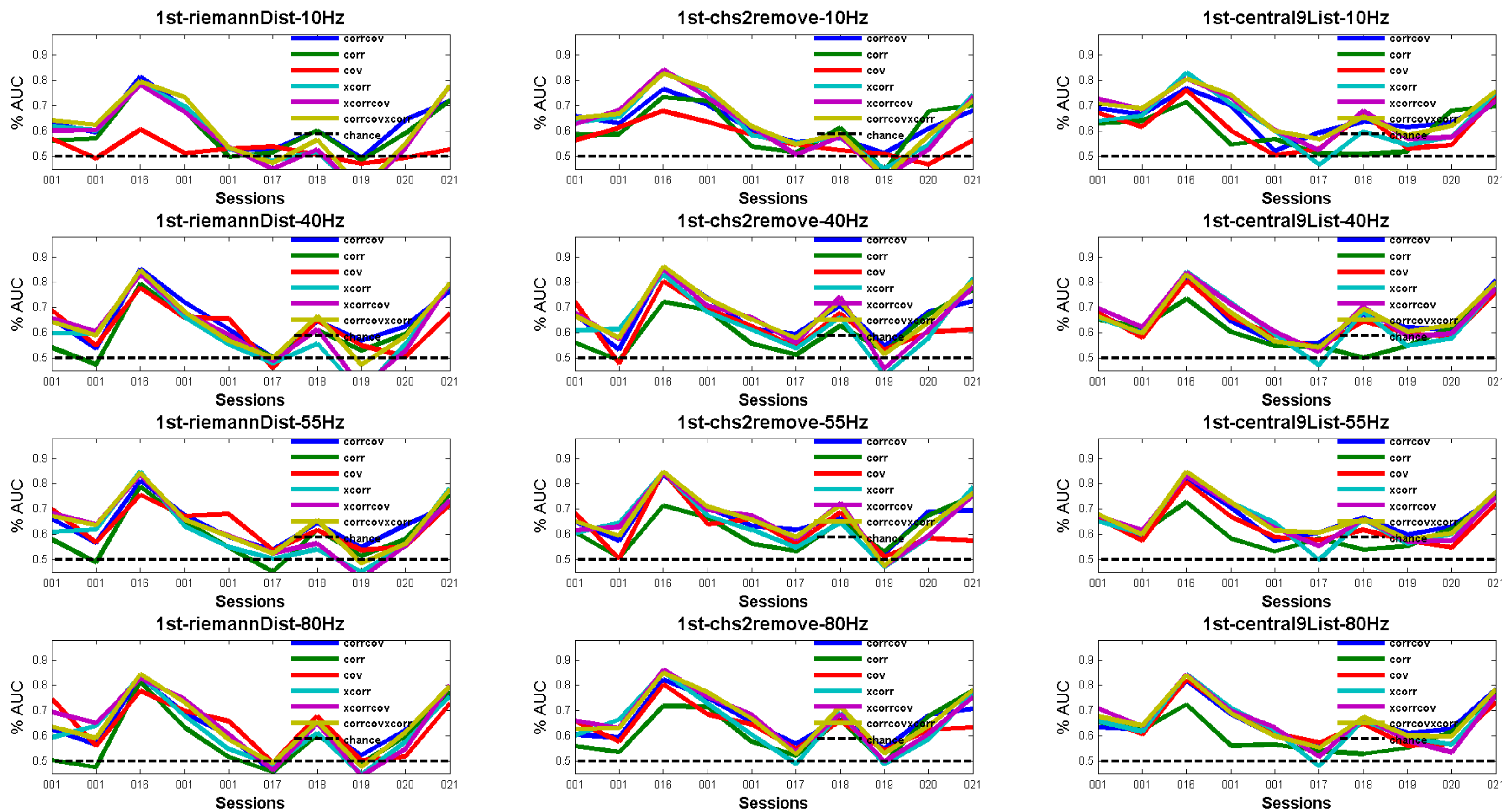


Figure III.17. AUC for different type of features, electrode selection function and band-pass frequencies for the oErrP across all closed-loop sessions.

AUC for multiple parameters evaluated with 10 iterations of 10-fold cross-validation (channels reduced with a 5^{th} order xDAWN filter and features z-scored). Y axis: AUC. X axis: closed-loop sessions. From top to bottom: 1-10, 1-40, 1-55, and 1-80Hz bandpass filter. From left to right: Riemann distance, Chs2Remove, and Central9List electrode selection function. Color traces represent different feature extraction function: *corrcov* (blue), *corr* (green), *cov* (red), *xcorr* (light blue), *xcorrcov* (magenta), *corrcovxcorr* (yellow). Dashed line represents chance. Subject 019 presented low performance values. The *corrcovxcorr* feature filtered 1-40Hz with the Chs2Remove electrode selection function provided the highest AUC (0.695).

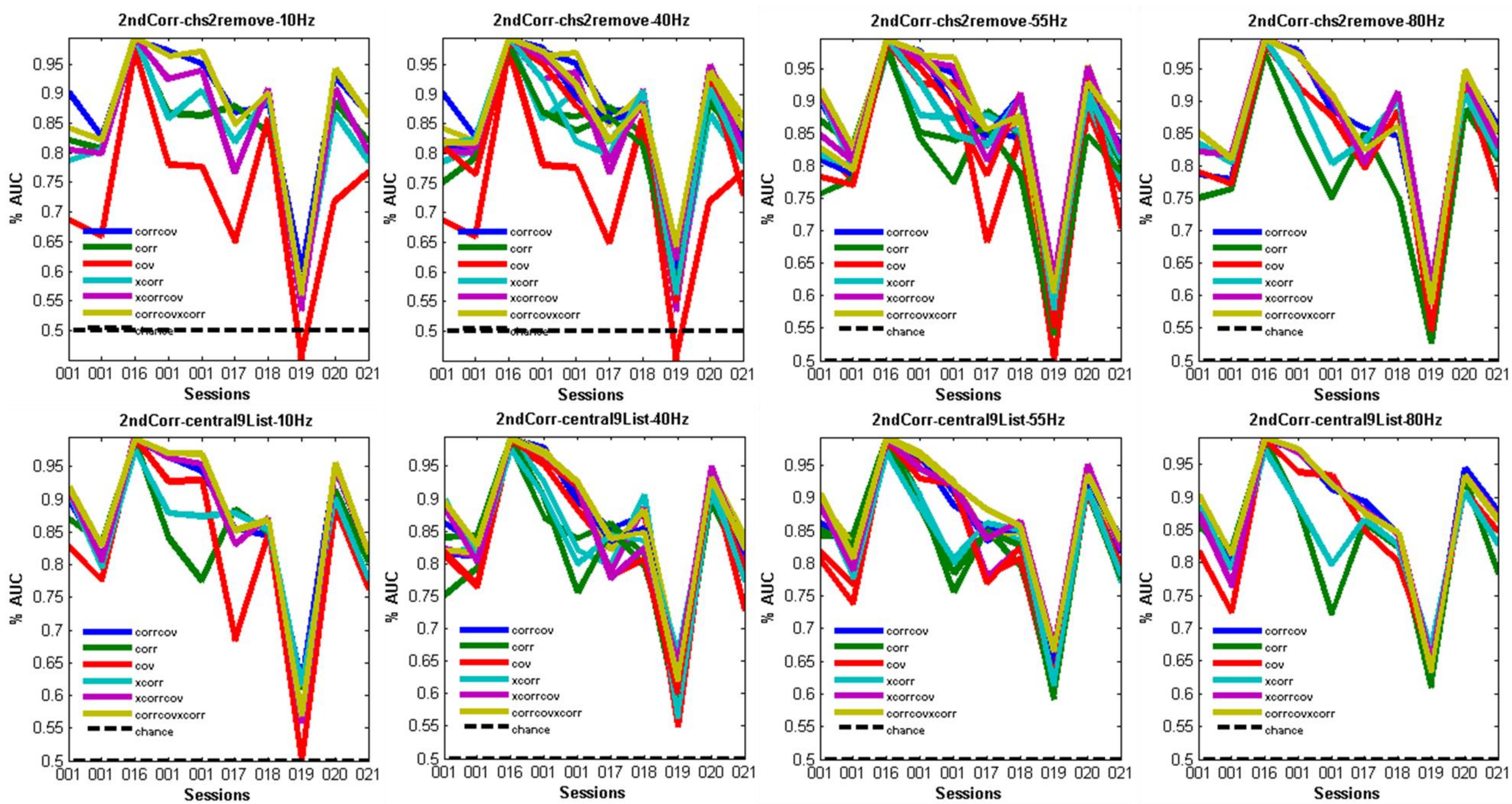


Figure III.18. AUC for different type of features, electrode selection function and band-pass frequencies for the iErrP (2ndCorr) across all closed-loop sessions.

AUC for multiple parameters evaluated with 10 iterations of a 10-fold cross-validation (channels reduced using a xDAWN filter of 5th order and features z-scored). The Y and X axes represent the AUC and the closed-loop sessions (2ndCorr subset), respectively. From left to right: 1-10, 1-40, 1-55, and 1-80Hz bandpass filter. From top to bottom: Chs2Remove and Central9List electrode selection function. Color traces represent different feature extraction function: *corrcov* (blue), *corr* (green), *cov* (red), *xcorr* (light blue), *xcorrcov* (magenta), *corrcovxcorr* (yellow). The dashed line represents chance (black). Most of the subjects present performance values above 0.8 in contrast to subject 019 whom shows poor performance values.

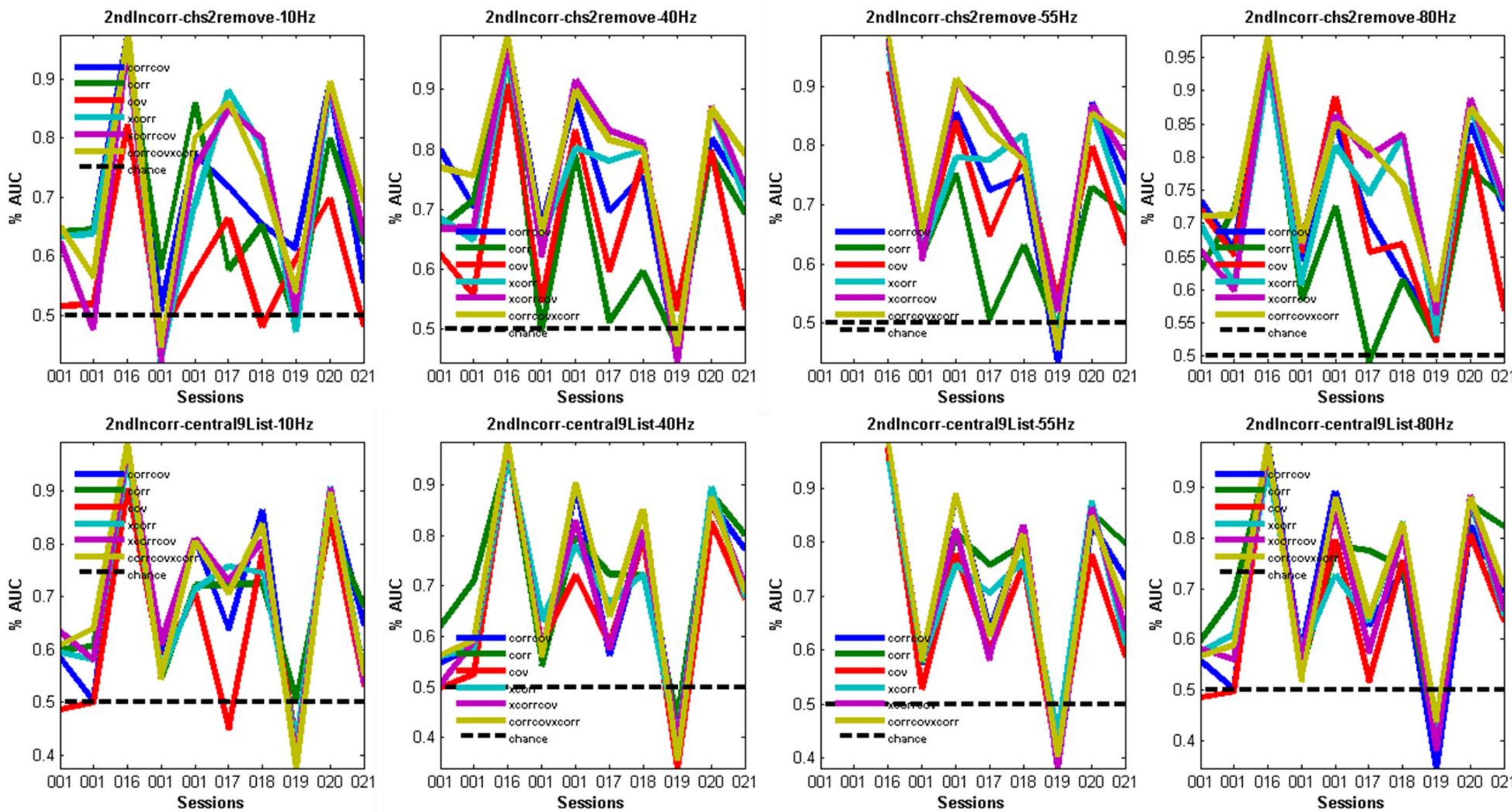


Figure III.19. AUC for different type of features, electrode selection function and band-pass frequencies for the iErrP (2ndIncorr) across all closed-loop sessions.

AUC for multiple parameters evaluated with 10 iterations of a 10-fold cross-validation (channels reduced using a xDAWN filter of 5th order and features z-scored). The Y and X axes represent the AUC and the closed-loop sessions (2ndIncorr subset), respectively. From left to right: 1-10, 1-40, 1-55, and 1-80Hz bandpass filter. From top to bottom: Chs2Remove and Central9List electrode selection function. Color traces represent different feature extraction function: *corrcov* (blue), *corr* (green), *cov* (red), *xcorr* (light blue), *xcorrcov* (magenta), *corrcovxcorr* (yellow). The dashed line represents chance (black). Overall, performance values are better than the ones obtained with the oErrP but not as good as the ones from the 2ndCorr data. Subject 019 continues presenting poor performance values (<0.6).

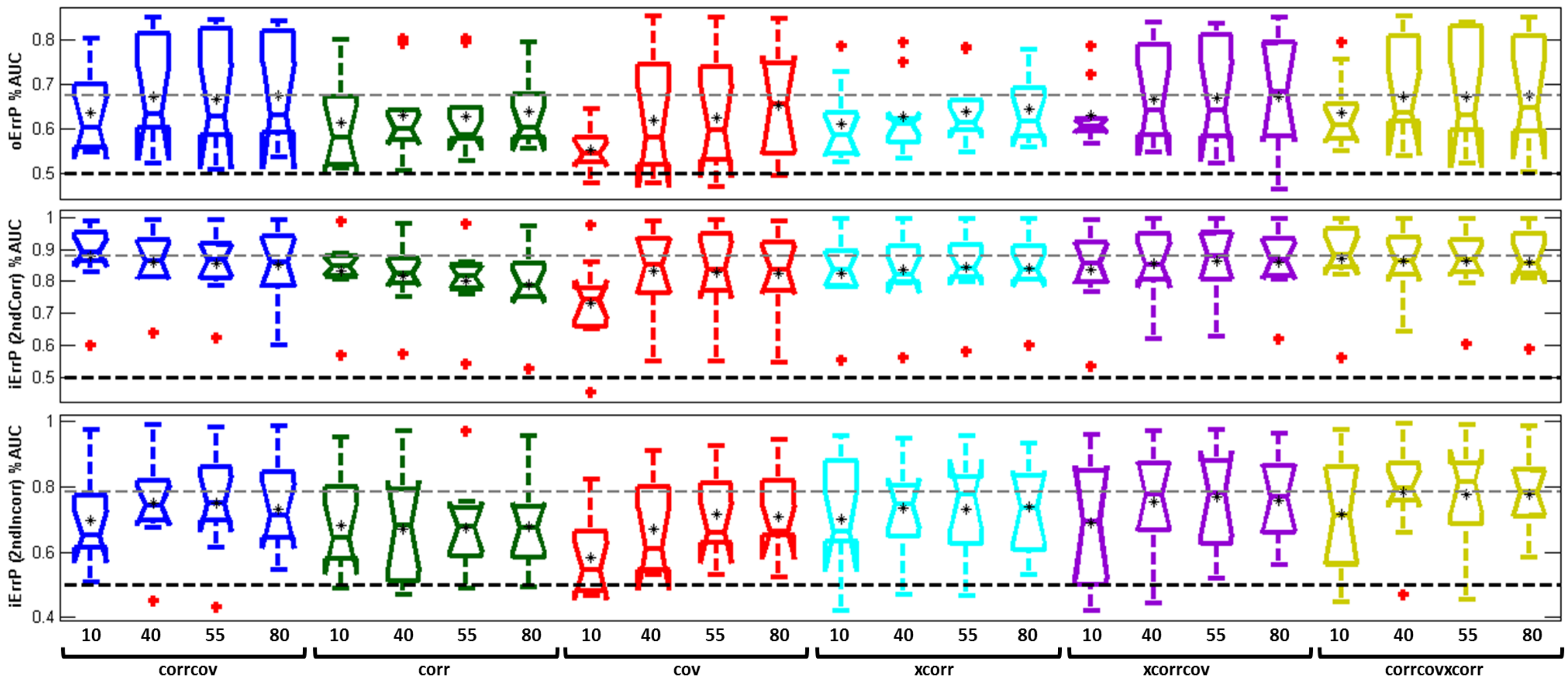


Figure III.20. Population AUC for different type of features and band-pass frequencies for the oErrP, iErrP (2ndCorr) and iErrP (2ndIncorr).

Box plots of the population AUC (median, quartiles and outliers) for multiple parameters evaluated with 10 iterations of 10-fold cross-validation (channels reduced using a 5th order xDAWN filter and selected using the Chs2Remove function; features z-scored. From top to bottom: oErrP (open- and closed-loop), iErrP 2ndCorr and iErrP 2ndIncorr (closed-loop only), respectively. Y axis: AUC. X axis: different passband filters and feature extraction functions. The passband from left to right are 1-10, 1-40, 1-55, and 1-80Hz. These features are repeated in the same order in each color group. Colors represent the feature extraction functions: *corrcov* (blue), *corr* (green), *cov* (red), *xcorr* (light blue), *xcorrcov* (magenta), *corrcovxcorr* (yellow). Black dashed-lines mark chance. The gray dashed-line marks the highest mean AUC value. The parameters for the best across-subject performance values are: *corrcov*, 1-10Hz filtered for the oErrP; *corrcov* and 1-10Hz filtered for the iErrP 2ndCorr subset; and *corrcovxcorr* and 1-40Hz filtered for the iErrP 2ndIncorr subset. Red crosses represent outlier values. Overall the 1-40Hz passband filter and the *corrcovxcorr* feature extraction function (yellow) presents the highest AUC for all ErrP. The *corr* (green) and xcorr (light blue) feature extraction functions showed the lowest performance AUC values.

3.5 Online ErrP detection (reaching task)

Figure III.21 presents the AUC performances for each of the different subgroups of naïve subjects (Closed-loop online, Closed-loop offline, Open-loop offline, and Secondary error). In the offline analysis, all channels were reduced using a 5^{th} order xDAWN filter per class, the features were z-scored, and no re-referencing was applied. The offline versions of the task had higher performance values than the online closed-loop task because the latter used only the available data from previous blocks, resulting in fewer observations for training. Closed-loop offline sessions presented an average AUC of 0.6219 ±0.0935 S.D. (with the chance level equal to 0.507; computed using the same data but with the labels shuffled); open-loop sessions showed an average AUC of 0.5940 ±0.0378 S.D. (with chance equal to 0.497); secondary ErrP (II + CI, 2ndIncorr + 2ndCorr) gave an average AUC of 0.7315 ± 0.1123 (with chance levels of 0.4953); secondary ErrP (II, 2ndIncorr) gave an average AUC of 0.7505 ± 0.0516 (with chance levels of 0.4833); and secondary ErrP (CI, 2ndCorr) gave an average AUC of 0.8712 ± 0.0430 (with chance levels of 0.5090). Shown at the far right of the figure (in green) is the AUC performance gain achieved by using classification of the iErrP (2ndCorr). This gain significantly increases the AUC to values > 0.8 for all naïve subjects (not including the meditation subject).

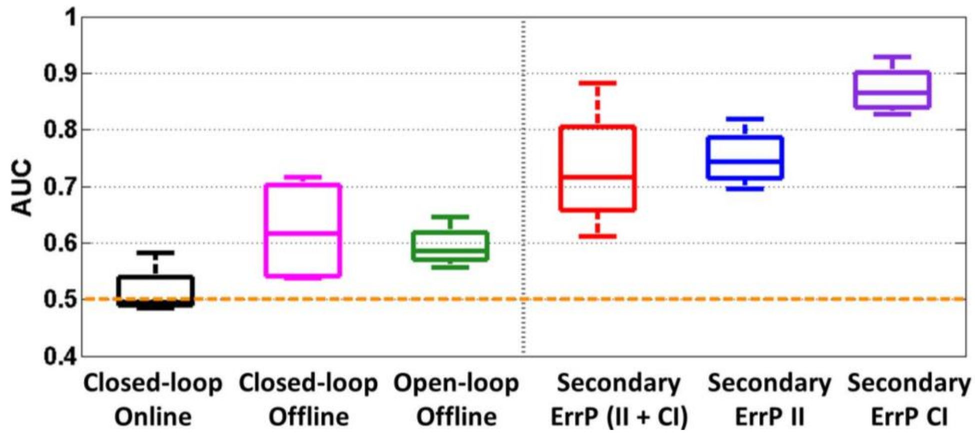


Figure III.21. Overall performance from the naïve (without meditation session) subjects under different analysis conditions.

Median and quartile distribution for the AUC performance values for ErrP classification of 4 naïve subjects (not including the meditation subject). Left to right: closed-loop online (black), closed-loop offline (magenta), open-loop offline (cyan), secondary Error (green). Closed-loop online values represent the actual performance of ErrP detection during the test closed-loop sessions (retraining the decoder at the end of each block). Closed-loop offline represents offline analysis performed with all the testing data and with parameters optimized. Open-loop offline represents the results obtained during the offline analysis (from data collected in the open-loop reaching task sessions). Secondary error results were obtained using both 2ndCorr and 2ndIncorr data (CI + II), 2ndIncorr (II) and 2ndCorr (CI) data from the closed-loop sessions and by performing offline ErrP detection (optimizing parameters). iErrP (CI 2ndCorr) classification outperforms all the other groups.

Figure III.22 further breaks down the classifier performance into True Positive Rates (TP), True Negative Rates (TN), False Positive Rates (FP) and False Negative Rates (FN) for each of the six analysis conditions. The FP in the closed-loop online sessions seems to highly decrease the overall performance. Offline analyses using the secondary ErrP (CI, 2ndCorr) for classification showed better true positive and true negative rates. Thus, implementing iErrP detection can provide a promising approach towards a reliable feedback system between humans and robots.

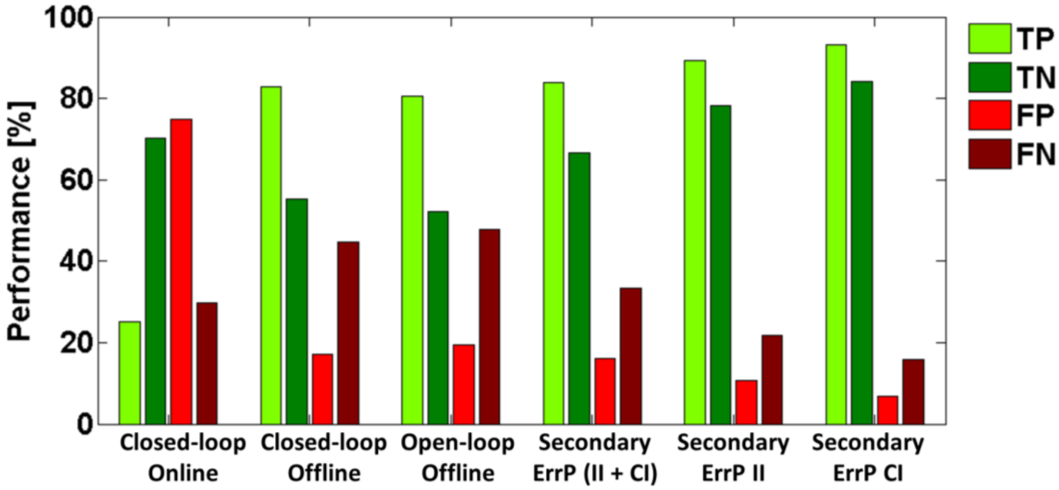


Figure III.22. Comparison of true-positive and true-negative values for the secondary ErrP (naive subjects, not including the subject meditating).

Class-specific decoder performance for the naïve subjects (not including the subject meditating): true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) are shown in light green, dark green, red and dark red, respectively. From left to right: closed-loop online, closed-loop offline, open-loop offline, offline secondary ErrP (II + CI, for 2ndCorr and 2ndIncorr data pooled together), offline secondary ErrP II (for 2ndIncorr data) and offline secondary ErrP CI (for 2ndCorr data). Closed-loop online values represents the actual performance of ErrP detection during the test closed-loop sessions. Closed-loop offline represents offline analysis performed with all the testing data. Open-loop offline represents the results obtained during the offline analysis. Secondary ErrP results were obtained using data from the closed-loop sessions and performing offline secondary ErrP detection. The TP and TN of the Secondary ErrP CI (2ndCorr) dataset outperform the values in the other groups.

3.6 Effect of meditation in the production of ErrP

Subject M (019) presented the lowest classification performance, both offline and online (across all subjects). To explore the effect meditation could have on ErrP classification performance, and on the evoked ErrP, we compared subject M's EEG signals with that of the other naïve subjects. We compared the average correct, average incorrect, and ErrDiff traces from the observation and interaction ErrP (2ndCorr and 2ndIncorr).

For the observation ErrP (Figure III.23), the mean correct, mean incorrect, and ErrDiff traces from subject M had smaller amplitudes than the traces from the other naïve subjects. The ErrDiff trace from subject M did not present a positive peak that was found in all the other subject's ErrDiff traces. For the iErrP (2ndCorr) the average incorrect and ErrDiff traces in subject M presented smaller amplitudes (Figure III.24). The average correct traces of all subjects were relatively similar in amplitude. Finally, while all the naïve subjects' average incorrect traces presented clear peaks in the signal, subject M's incorrect trace was rather flat. For the iErrP (2ndIncorr), although only a small number of trials was available for computing the average traces (see

Table III.1), a decrease in the average correct trace was found (not in the incorrect and ErrDiff traces). For iErrP (2ndIncorr) trace comparison refer to the Appendix.

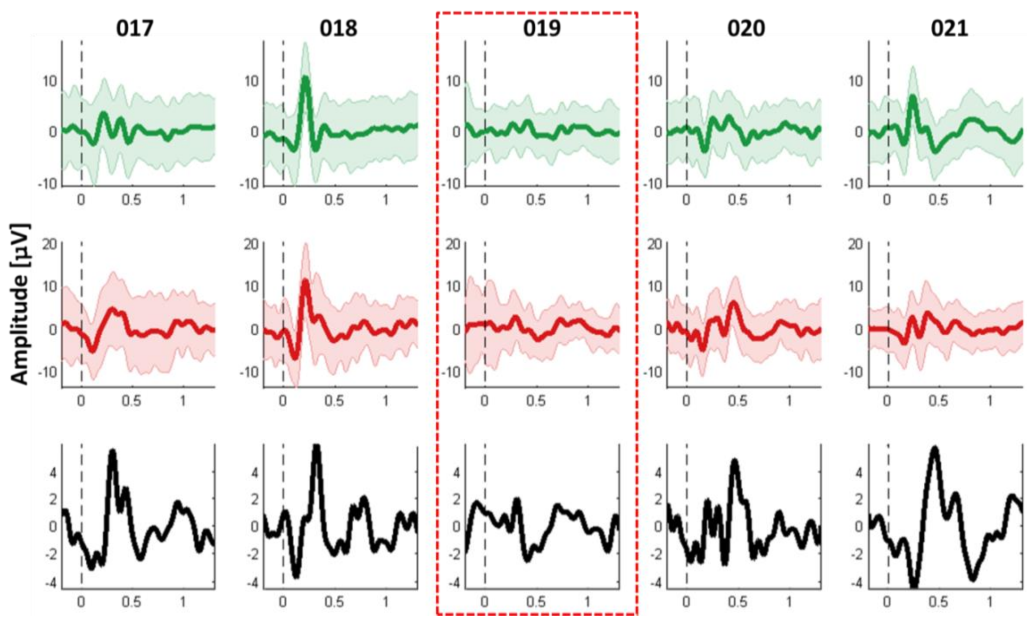


Figure III.23. Average correct, incorrect and ErrDiff traces from all naive subjects (oErrP).

Average traces for the trials representing correct and incorrect Baxter target selection, and their ErrDiff, for all naïve subjects (during closed-loop online sessions). Columns represent the subject number; from left to right: 017, 018, 019, 020, 21.The correct (green), incorrect(red) and ErrDiff traces (oErrP) belong to electrode Cz. Error bars depict standard deviation of correct (light green) and incorrect (light red) trials. The X axis is the time to feedback onset (seconds), and the Y axis represents signal amplitude (µV). The red dashed-square surrounds subject M's traces. All traces from each column have the same scale. The average correct, average incorrect and ErrDiff traces from subject M are smaller in amplitude than the same traces from the rest of subjects.

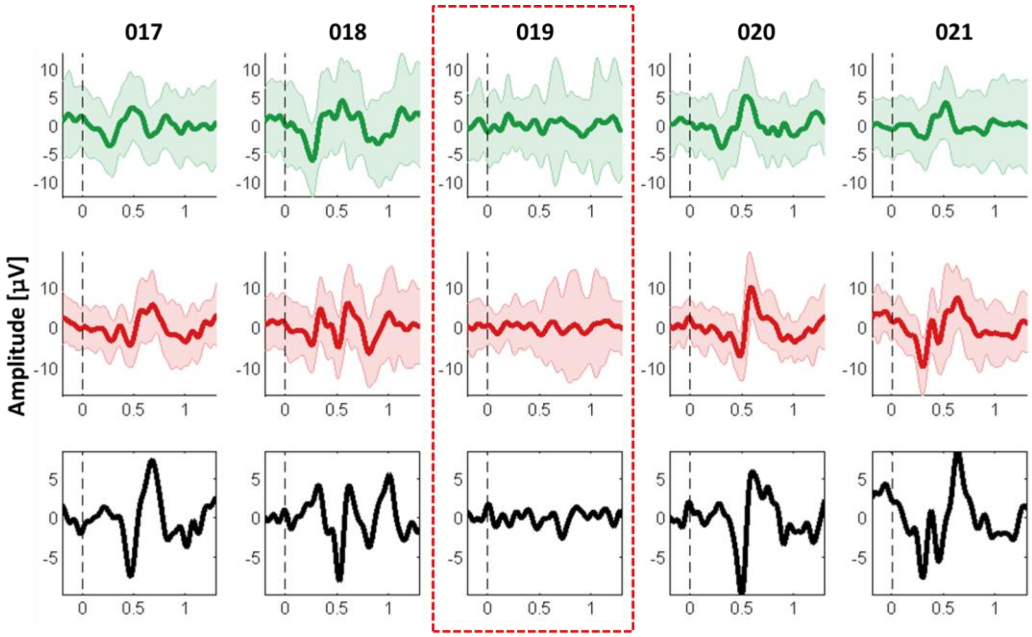


Figure III.24. Average correct, incorrect and ErrDiff traces from all naive subjects (iErrP 2ndCorr).

Average traces for the ErrP decoder outcome class and their ErrDiff for all naïve subjects (after a correct selection by Baxter, 2ndCorr). Columns represent the subject number; from left to right: 017, 018, 019, 020, 21.The correct (green), incorrect(red) and ErrDiff traces (iErrP) belong to electrode Cz. Error bars depict standard deviation of correct (light green) and incorrect (light red) trials. The X axis is the time to feedback onset (seconds), and the Y axis represents signal amplitude (µV). The red dashed-square presents subject M (019) traces. All traces from each column have the same scale. The average incorrect and ErrDiff traces from subject M are smaller in amplitude than the same traces from the rest of subjects.

4. Discussion

In this chapter we presented, to our knowledge, the first real-time BMI that relies solely on ErrP for directly detecting volitional commands. The ErrP signals were employed as an indicator of the user's expectations under a binary-choice task. By focusing on the detection of naturally occurring ErrP, an online closed-loop EEG system that enables intuitive human-robot interaction was developed. This system was tested with general population subjects that had not been previously trained on the task or with EEG systems. The presence of oErrP and iErrP was confirmed, and their suitability for single-trial error detection was explored. Our results suggest that interactive ErrP are easier to classify (with average AUC performance values that can reach above 0.8) and that they should be used in the future for closed-loop classification.

4.1 Binary-choice still images task

Determining the feasibility of controlling a robot arm for real-time single-trial ErrP decoding initially required confirmation that ErrP signals could be evoked during robot arm reaches. With this purpose we ran a preliminary study focused on confirming the presence of ErrP in a binary-choice task. The binary-choice images paradigm was developed and tested (using images of Baxter performing a reach), using a photodiode to properly signal the feedback onset time. The timing and location of the negative and positive peaks (in the range of 200ms and 350-400ms post-feedback onset, mainly in the central electrodes Cz and FCz) of the epoched data follows what has been reported in the literature for observation ErrP (Chavarriaga et al., 2014; Chavarriaga et al., 2007).

4.2 Binary-choice reaching task

With the ErrP characterized we moved to design the binary-choice reaching paradigm and found several problems with using the experiment codes to signal the feedback onset time. Time synchrony is essential for finding the ErrP, especially for single-trial classification. Although new methods are being explored to take into account latencies in the signal (Iturrate et al., 2014), this is not straight forward for real-time classification. Data recorded during the reaching task revealed that the experiment codes sent by the Experiment Controller presented a delay, in some cases of up to 500ms. These delays were caused by the operating system (by the task scheduler) of the Experiment

Controller computer and by the analysis loop of the EEG/Robot Interface. The pushbutton switch, a single-bit hardware experiment code, was employed to bypass these software delays. The tests and simulations of the pushbutton behavior proved to provide reliable feedback onset times.

With a proper method to signal the feedback onset time we analyzed the observation ErrP evoked by the incorrect robot reaches across the open-loop sessions. The times of the negative and positive peaks, and their location in the electrode layout, agreed with the values found in the literature: negative peaks occurring near 250ms post-feedback onset in the fronto-central electrodes and positive peaks happening between 350ms and 450ms post-feedback onset with a centro-parietal distribution (Ferrez & Millan, 2005; Perrin et al., 2010; Chavarriaga et al., 2007).

To properly analyze these traces, bad trials and channels with eye movements had to be removed. A semi-automatic technique using the mean and standard deviation of the traces was used to detect and discard bad trials. Also, the Chs2Remove electrode selection function was used to remove all the electrodes close to the eyes, the masseter and temporal muscles, and the occipital electrodes (Iturrate et al., 2010). This approach proved to help the decoder performance by eliminating possible sources of noise.

4.3 Offline analysis

Before running the closed-loop sessions, offline analysis suggested that the highest ErrP classification performance values were obtained with the following parameters: applying a CAR filter over each of the Central9List electrodes, using 5th order xDAWN filters, extracting the *cov* and *corr* features (*covcorr*), and z-scoring the feature vectors. Amongst the different classification algorithms explored, the linear regression with Elastic Net shrinkage produced the best decoding values. Better decoder performance was found with the Riemann distance and the Chs2Remove electrode selection functions but the Central9List function was chosen instead because it allowed small covariance matrices. Tangent space mapping of large covariance matrices (especially at the eigenvalue decomposition step) usually took more than 400ms during single-trial classification, hence affecting the real-time functionality of the decoder.

Also, we used the AUC performance metric to account for the unbalanced number of trials in some of the sessions (when the correct/incorrect number of trials was not similar). The use of classification accuracy could have masked the true chance level. Overall, subjects 014 and 016 showed high ErrP classification values, regardless of techniques, filters and features extracted. Subject M presented the lowest performance in the three scenarios analyzed (oErrP, iErrP 2ndCorr, and iErrP 2ndIncorr).

Although preliminary (only from one subject), the results from subject M suggest interesting ideas worth exploring in the future. Initially, our hypothesis was that a person in a meditative state would be better suited for guiding his goal-directed behavior, including focusing his attention and filtering out any sources of noise or distraction. We also thought this subject would be more engaged in the task. Our results suggest otherwise. The amplitudes of subject M’s average correct, average incorrect and ErrDiff traces belonging to oErrP were smaller than the amplitudes of the traces found in the other naïve subjects. The same was true for the average incorrect and the ErrDiff traces from iErrP 2ndCorr. Moreover, the average incorrect traces in the iErrP 2ndCorr suggested that subject M did not strongly react to the classification errors.

Meditation has been mainly classified in two different styles: Focused Attention (FA) and Open Monitoring (OM) meditation. FA entails the voluntary focusing of attention on a chosen object. OM meditation involves non-reactive monitoring of the content of experience from moment to moment (Lutz et al., 2008; Manna et al., 2010). Initially, we thought our subject M would be focusing his attention and filtering out all sources of noise; hence increases the error-related signal. The results suggest subject M was performing OM meditation: no external cues triggered changes in his mental states, nor increased his reactivity to errors; his attentional levels were not affected by the external world, not only the noise and distraction but also Baxter's and the decoder's errors. This would explain the lack of a high amplitude ErrDiff signal and the low amplitude of both average correct and incorrect trace, especially in the iErrP 2ndCorr. One possible explanation for this phenomenon is that meditation decreases the gain in the attention boost that an error is supposed to trigger (Ullsperger et al., 2014). For this reason, only the iErrP 2ndCorr incorrect trials have smaller amplitudes whereas the correct trials present similar amplitude values to the ones found in other naïve subjects. This is true if we assume that the magnitude of the ErrP signal encodes such attention gain. These ideas could be explored in a larger population of people that meditate (and tested before, during and after a meditative state).

4.4 Closed-loop online performance

We provided a proof of concept of single-trial real-time ErrP classification. Once trained offline using a small sample of open-loop trials, our pipeline can decode brain signals fast enough to be used in real-time. Offline analysis of the closed-loop data demonstrated that iErrP are typically easier to classify than oErrP, and can thus be used to improve the performance accuracy. For the closed-loop online sessions only data from the previous blocks were used for training. This approach seems to affect the overall decoder performance of the session since the first blocks had very few observations to train the decoder. More data allows better generalization of any decoding algorithm. This is one of the reasons our closed-loop online results were so close to chance and did not reach similar performance values as in the offline analysis (AUC = 0.65). Similar result values have been presented in complex scenarios beyond the lab environment, such as detecting ErrP signals during a real-world driving task (Zhang et al., 2015).

Moreover, the algorithms implemented in the online classification pipeline need to be improved since the code used in Simulink was different to the one used for offline analysis. This made comparisons across classification pipelines (offline vs. online) cumbersome and prone to errors. Also, the computations required to implement the xDAWN filtering in real-time restricted the number of electrodes used for ErrP classification. The implementation of the spatial filtering and the online ErrP classification in a more powerful machine might allow, in the future, the use of more robust and computationally complex algorithm, which may improve classification performance.

Beyond the hardware and software bottlenecks, the difference in the offline decoder performance of the oErrP and iErrP reveals that true task engagement can strongly affect the BMI performance (AUC increasing from 0.65 up to 0.74). Interaction ErrP require an active participation in the task. In the future the subjects should be informed they are controlling the system (even in the first target selection) to provide a sense of agency and control of the events happening in the task. Being a mere observer decreases the difference in the signal amplitude and classification performance. Moreover, iErrP classification can be incorporated into the online scenario to boost closed-loop performance as well (after oErrP has been classified). This suggests that new paradigms can be designed to exploit the iErrP for multiple-choice tasks or even continuous control. For example, the robot can perform motions that are designed to elicit error potentials from the user in order to acquire feedback at crucial times when choosing between many possible options or even adjusting a continuous trajectory in real time. Implementing these changes and suggestions may improve the classification performance in the future. This way, the presented system moves closer towards the goal of creating a framework for intuitive human-robot interaction in real-world tasks.

5. Appendix

5.1 Riemann Geometry Equations

5.1.1 Riemann distance and Geometric Mean computation (Congedo et al., 2013; Barachant et al., 2012; Barachant & Congedo, 2014; Barachant et al., 2013)**.** Let $X \in R^{Ch\,x\,S\,x\,Nt}$ be the matrix comprised of all the trials (from both correct and incorrect classes), with *Ch* the number of channels, *S* the total number of samples, and *Nt* the total number of trials from both classes. Let $X_i^{(k)} \in R^{Ch\ x\ S}$ represent the signal for trial *i* belonging to class *k*, and y_i be the class value ($y_i = k$; $k \in \{0,1\}$, correct or incorrect, respectively) for trial *i*. We define the sample covariance matrix (SCM) for each trial *i* as:

$$C_i = \frac{1}{(S-1)} X_i X_i^T$$

Let $C^{(k)}$ represent the average covariance matrix (ACM) obtained from computing the Riemann geometric mean of all the class *k* covariance matrices.

The Riemann geometric mean $\Re(\mathrm{C}_1, \mathrm{C}_2, \dots, \mathrm{C}_\mathrm{j})$ is defined as:

$$C^{(k)} = \Re(C_1, C_2, \dots, C_j) = argmin_{\bar{C}} \sum_{j=1}^{T_k} \delta_R^2(C_j, \bar{C}),$$

with $T_k = \{j \mid y_j = k\}$ denoting the total number of trials belonging to class *k*, and $\bar{C}$ representing an estimator of the average covariance matrix (see Algorithm 1).

The Riemann distance δ_R between the sample covariance matrix, $\mathrm{C_i}$, and the class average covariance matrix, $\mathrm{C^{(k)}}$, is defined as the Frobenius norm of $\log(C_i^{-1}, C^{(k)})$:

$$\delta_R(C_i, C^{(k)}) = \left\|\log(C_i^{-1}, C^k)\right\|_F = \left[\sum_{j=1}^{Ch} log^2 \lambda_j\right]^{1/2},$$

where $\{\lambda_j\}_{j=1}^{Ch}$are the real eigenvalues of $C_i^{-1} C^{(k)}$.

The geometric mean is computed by an iterative procedure that comprises 1) projecting the sample covariance matrices (belonging to class *k*) in the tangent space, b) estimating the arithmetic mean of the projected SCM and c) projecting back this result into the Euclidean space (via exponential mapping). The algorithm employed to compute the geometric mean of the sample covariance matrices belonging to class *k* is:

Algorithm 1. Geometric Mean of T_k covariance matrices (Barachant et al., 2013)

Input: $\mathcal{M}$, a set of T_k covariance matrices such that all C_i belong to class k and $\epsilon > 0$.
Output: Υ the estimated geometric mean in $T_k(Ch)$ of $\mathcal{M}$.

1: Initialize $\Upsilon^{(1)} = \frac{1}{T_k}\sum_{i=1}^{T_k} C_i$ {Euclidean mean of the *T* covariances}

2: **repeat**

3: $\bar{S} = \frac{1}{T_k}\sum_{i=1}^{T_k} \boldsymbol{Log}_{\boldsymbol{\Upsilon(t)}}(C_i)$ {Arithmetic mean in the tangent space using $\mathfrak{H}^{(t)}$}

4: $\Upsilon^{(t+1)} = \boldsymbol{Exp}_{\boldsymbol{\Upsilon(t)}}(\bar{S})$ {Project back to Euclidean space using $\mathfrak{H}^{(t)}$}

5: **until** $\|\bar{S}\|_F < \epsilon$

6: **return** $\Upsilon^{(t+1)}$

Projecting a covariance matrix C_i into the tangent space involves its logarithmic mapping using a reference matrix C_{ref} such that:

$$S_i = \mathbf{Log}_{C_{ref}}(C_i) = C_{ref}^{1/2} \operatorname{logm}(C_{ref}^{-1/2} C_i C_{ref}^{-1/2}) C_{ref}^{1/2},$$

with C_{ref} representing the geometric mean, and Logm(.) denoting the logarithm of a matrix. The logarithm of a diagonalizable matrix $B = VDV^{-1}$ is defined as Logm(B) $= VD'V^{-1}$; with D' as the diagonal matrix and with its diagonal elements defined as

$$d'_{i,i} = \log(d_{i,i}).$$

Projecting from the tangent space back into the Euclidean space requires an exponential mapping of S_i via C_{ref}:

$$C_i = \mathbf{Exp}_{C_{ref}}(S_i) = C_{ref}^{1/2} \operatorname{expm}(C_{ref}^{-1/2} S_i C_{ref}^{-1/2}) C_{ref}^{1/2}$$

with Expm(.) denoting the exponential of a matrix. Its computation follows the same logic as with the Logm(.) operator, i.e. $d'_{i,i} = \exp(d_{i,i})$.

5.1.2 Minimum Distance to Riemann Geometric Mean (Barachant & Congedo, 2014)**.** The Minimum Distance to Riemann Mean classifier involves a supervised training phase. Given a training set of $X_i \in R^{Ch\ x\ S}$ trials and their corresponding labels $y_i \in \{1{:}\,k\}$, with $i \in \{1{:}\,Nt\}$, training involves the estimation of the sample covariance matrices C_i for each trial *i*, and the average covariance matrix $C^{(k)}$ for each class *k*:

$$C^{(k)} = \Re\left(C_j |\, y_j\ = k\right)$$

with $\Re(.)$ The Riemann geometric mean operator. The classification of a testing trial X_t belonging to an unknown class *y* is achieved computing the Riemann distance between C_t (SCM of X_t) and each of the average covariance matrices $C^{(k)}$. The estimated class will be the one providing the minimum Riemann distance across all classes.

$$\hat{y} = argmin_k\ \delta_R\ (C_t, C^k)$$

The complete algorithm is:

Algorithm 2. Minimum Dist. to Riemann Mean (Barachant & Congedo, 2014)

Input: a set of trials $X \in R^{Ch\ x\ S\ x\ Nt}$ belonging to *Nk* classes and the corresponding labels $y_i\ \in \{1{:}\,Nk\}$.

Input: X_t a test trial of unknown class *y*.

Output: $\hat{y}$ the estimated class of the test trial.

1: Compute C_i for each trial *i*. {sample covariance matrix}

2: Compute C_t for the unknown class trial. {sample covariance matrix}

3: **for** *k* =1 to *Nk* **do**

4: $\mathrm{C}^{(\mathrm{k})} = \Re(\mathrm{C_i} |\, \mathrm{y_i} = \mathrm{k})$ {Riemann Geometric mean per class}

5: **end for**

6: $\hat{y} = argmin_k\, \delta_R\left(C_t, C^{(k)}\right)$ {Riemann distance}

7: **return** $\hat{y}$

5.1.3 Electrode Selection using Riemann distance (Barachant & Bonnet, 2011)

Selecting the η electrodes that better allowed separation between both classes was performed finding the electrode list υ_{red} that maximized the Riemann distance between the average covariance matrices of both classes over those electrodes, $C^{(1)}(\upsilon_{red})$ and $C^{(2)}(\upsilon_{red})$. The algorithm is based on backwards selection starting from *Ch* electrodes until reaching η electrodes. Let the operator ℓ*(V,iCh)* represent removing from the vector *V* the element in the *iCh-th* location. The algorithm is defined as follows:

Algorithm 3. Electrode selection using Riemann Distance

Input: a set of trials $X \in R^{Ch\, x\, S\, x\, Nt}$ belonging to *2* classes and the corresponding labels $y_i \in \{1{:}2\}$.

Input: υ, the complete electrode list (1:Ch), and η, the desired maximum number of electrodes.

Output: υ_{red}, a vector with the indexes of the selected electrodes (reduced).

1: Initialize $\upsilon_{red} = \upsilon$ {start with all electrodes}

2: Initialize nυ = *Ch* {start with total number of electrodes}

3: Compute $\mathrm{C_i}$ for each trial *i*. {sample covariance matrix}

4: **for** k =1 to 2 **do**

$C^{(k)} = \Re(C_i |\ y_i = k)$ {Riemann Geometric mean per class}
end for
while nυ > η **do**
  $\mathrm{n}\upsilon = \mathrm{n}\upsilon - 1$ {Remove one electrode at a time}
  for *iCh* = 1 to $n\upsilon$ **do**
    subElec $= \upsilon_{red}$
    $subElec = \ell(subElec, iCh)$ {Remove *iCh* from $subElec$}
    $\mathrm{dist(iCh)} = \delta_{\mathrm{R}}\left(\mathrm{C}^{(1)}(\mathrm{subE}, \mathrm{subE}), \mathrm{C}^{(2)}(\mathrm{subE}, \mathrm{subE})\right)$
  end for
  $\dot{\iota Ch} = \mathrm{argmax}_{iCh}$ dist*(iCh)* {Electrode with Max. Riemann dist.}
  $\upsilon_{red} = \ell(\upsilon_{red}, \dot{\iota Ch})$ {Remove $\dot{\iota Ch}$ from υ_{red}}
end while
return υ_{red}

5.2 xDAWN Equations

5.2.1 xDAWN filter (Barachant, 2014; Rivet et al., 2009; Rivet et al., 2011). Let $X_i \in R^{Ch\ x\ S}$ represent the *i*th trial, with *Ch* the number of channels and *S* the number of time samples. Let $P^{(k)}$ be the class average for class *k*.

$$P^{(k)} = \frac{1}{|Ind^{(k)}|} \sum_{i\ \epsilon\ Ind^{(k)}} X_i,$$

with y_i the class value ($y_i = k$; $k \in \{0,1\}$, correct or incorrect) for trial *i*, and $Ind^{(k)}$ the set of indices for the trials belonging to class *k*. $Ind^{(k)} = \{i \mid y_i = k\}$. Let X_o be the matrix with the all trials, for all classes, comprised of all the trials from both classes; $X_o \in R^{Ch\ x\ S\ x\ Nt}$, with *Nt* the total number of trials from both classes. Let $X \in R^{Ch\ x\ S*Nt}$, represent the two-dimensional matrix with all the trials concatenated per channel.

Let $w_1^{(k)}$ be an xDAWN (spatial filter) for class *k*, with $w_1^{(k)} \in R^{Ch\ x\ 1}$. The spatial filter is estimated to increase the signal to signal-to-noise ratio of a given class. For class *k*, *w* is defined as:

$$w = argmax_w \frac{w^T P^{(k)} P^{(k)^T} w}{w^T X X^T w}$$

This equation is a generalized Rayleigh quotient and can be solved using the eigenvalue/eigenvector decomposition of matrix $[(\mathrm{P}^{(\mathrm{k})}\mathrm{P}^{(\mathrm{k})^{\mathrm{T}}})(\mathrm{XX}^{\mathrm{T}})^{-1}]$. The solution to this generalized eigenvalue problem provides C solutions, ranked by the value of its eigenvalues. For each class *k*, we selected the D eigenvectors (D best spatial filters), corresponding to the D highest eigenvalues (with $D \epsilon \{2{:}\,5\}$). Each selected eigenvector is normalized by its L2-norm:

$$W_d^{(k)} = \frac{W_d^{(k)}}{\left\|W_d^{(k)}\right\|_2},$$

where $d \in \{1{:}\,D\}$. Let $W^{\{k\}} \in R^{Ch\ x\ D}$ be the selected spatial filters for class *k*, and $W \in R^{Ch\ x\ 2D}$ the aggregated matrix of the spatial filters for both classes, $W = [W^{(0)} W^{(1)}]$. The spatial filtering performed by the xDAWN filter is a linear projection of the average class by the matrix *W*:

$$Z_k = W^T P^{(k)}$$

Let Z_k represent the new xDAWN spatially filtered average trial for class *k*. Let Z_0 and Z_1 be corrxDAWN and incorrxDAWN , respectively.

5.2.2 Covariance xDAWN features (Barachant & Congedo, 2014; Barachant, 2014)**.** Let $X_i \in R^{Ch \, x \, S}$ represent the i^{th} trial, with *Ch* the number of channels and *S* the number of time samples, and $Z_\text{k} \in R^{D \, x \, S}$ the xDAWN filtered class *k* average (Z_0 and Z_1 corrxDAWN and incorrxDAWN , respectively) with *D* the number of filters per class. Let $zX_i \in R^{(2D+Ch) \, x \, S}$ be the augmented trial by appending to X_i each the spatially filtered average trial for both classes:

$$zX_i = \begin{bmatrix} Z_0 \\ Z_1 \\ X_i \end{bmatrix},$$

Let $zC_i \in R^{(2D+Ch) \, x \, (2D+Ch)}$ represent the covariance of the augmented i^{th} trial:

$$zC_i = \frac{1}{(S-1)} zX_i zX_i^T$$

From a set of *Ntrain* augmented covariance matrices zC_i, the tangent space mapping yielded a set of *Ntrain* vectors $cov_i \in R^{(2D+Ch+1)(2D+Ch)/2 \, x \, 1}$ (Barachant et al., 2013). This tangent space projection uses the geometric mean from all the *Ntrain* zC_i trials, zC_{ref}. The resulting matrices are vectorized (extracting only the upper elements of each projected matrix). Let zC_{ref} be defined as:

$$zC_{ref} = \Re(zC_1, zC_2, \dots, zC_{Ntrain}),$$

with the $\Re(.)$ operator representing the Riemann geometric mean. The tangent space projection of each zC_i trial *i* is defined as:

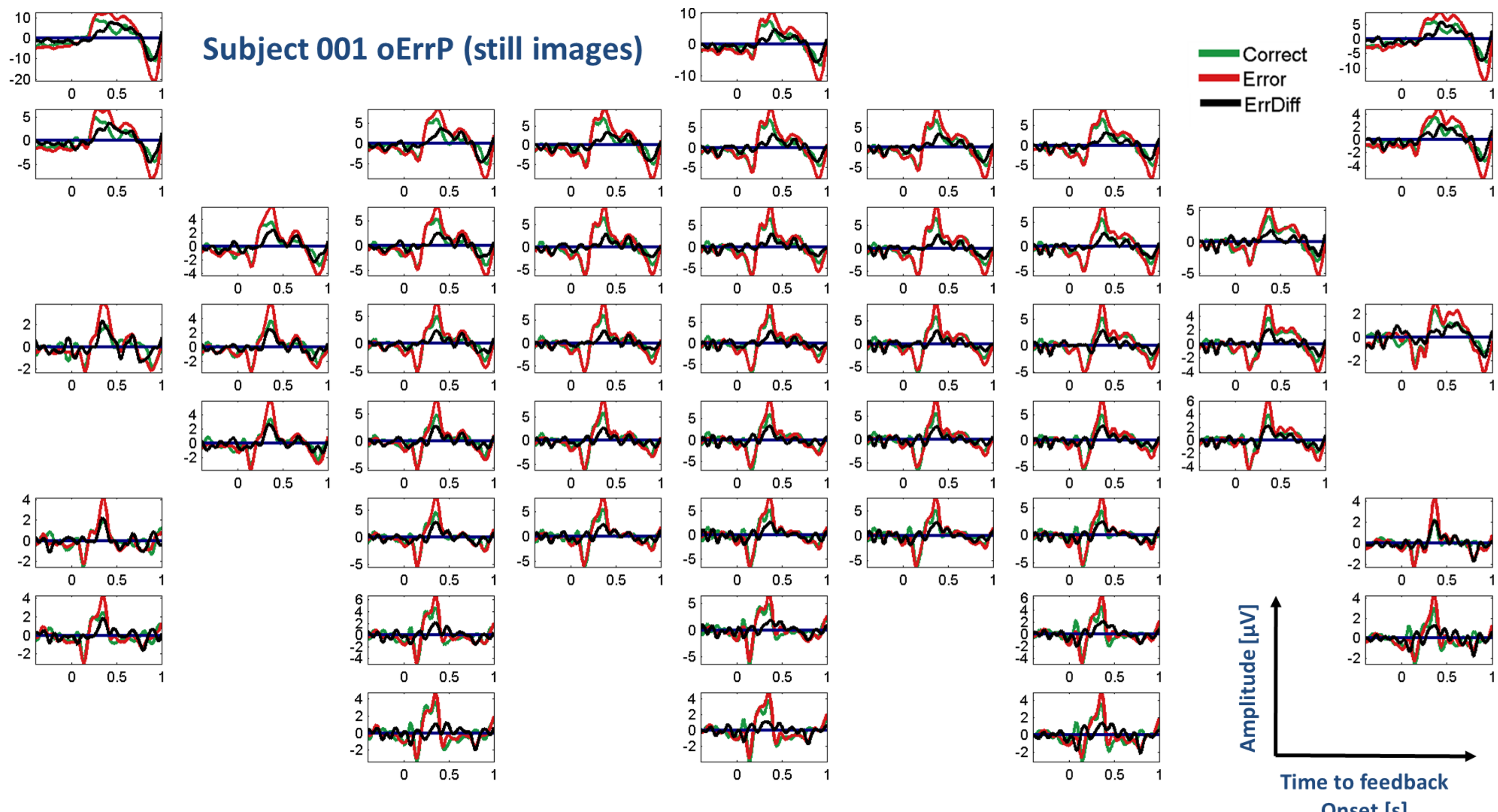
$$S_i = \mathbf{Log}_{zC_{ref}}(zC_i)$$

Due to symmetry, the projected S_i matrix is vectorized by applying a half-vectorization operator $vect(.)$ that concatenates, with appropriate weighting (with a $\sqrt{2}$

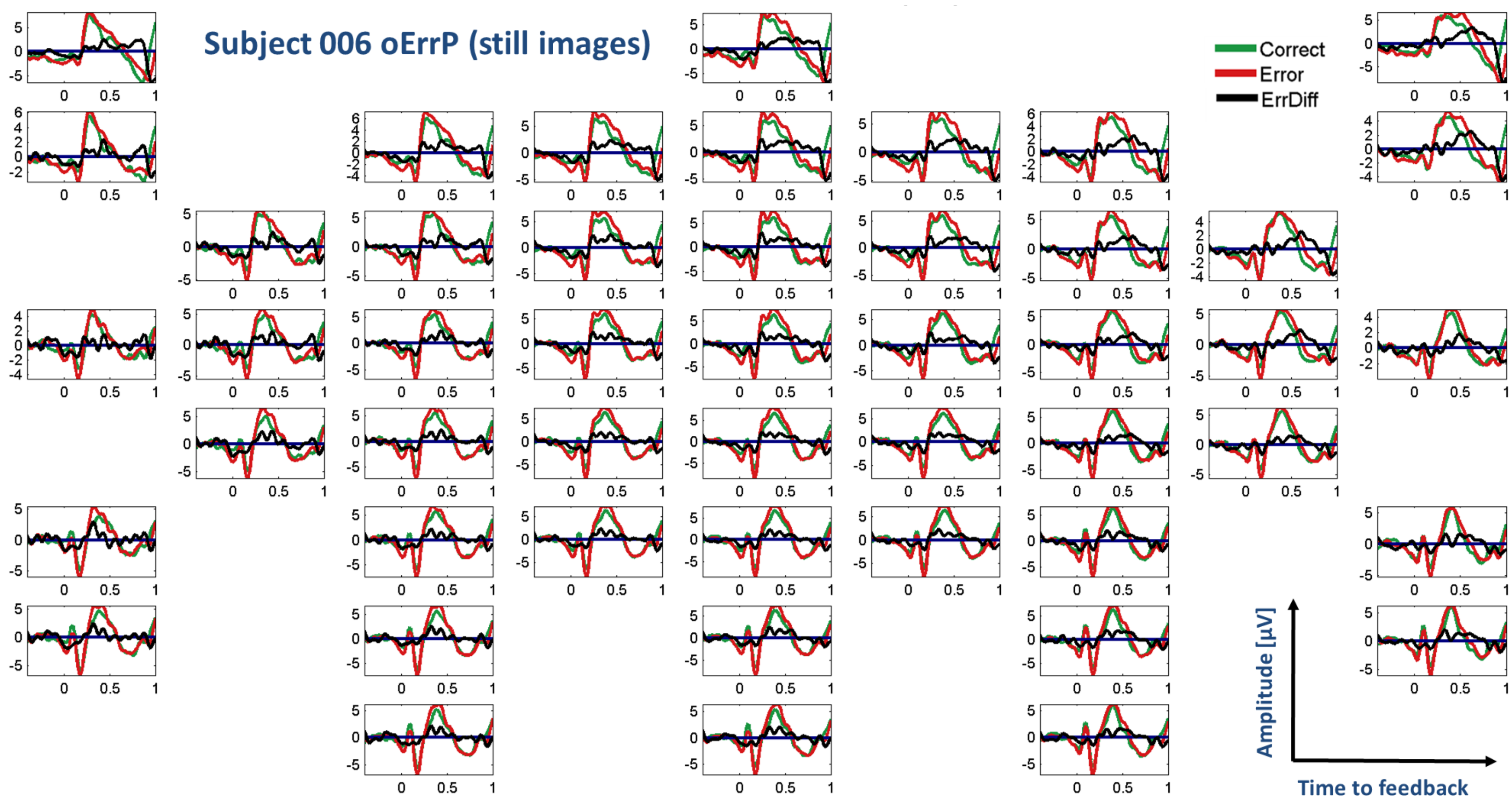
coefficient applied to the non-diagonal elements in order to conserve equality of norms), the upper triangular part of S_i. This results in $(2D + Ch + 1)(2D + Ch)/2$ features per cov_i vector:

$$cov_i = vect(S_i) = \left[S_{i\,1,1}, \sqrt{2}S_{i\,1,2}, S_{i\,2,2}, \sqrt{2}S_{i\,1,3}, \sqrt{2}S_{i\,2,3}, S_{i\,3,3}, \ldots, S_{i\,2D+Ch,2D+Ch}\right]$$

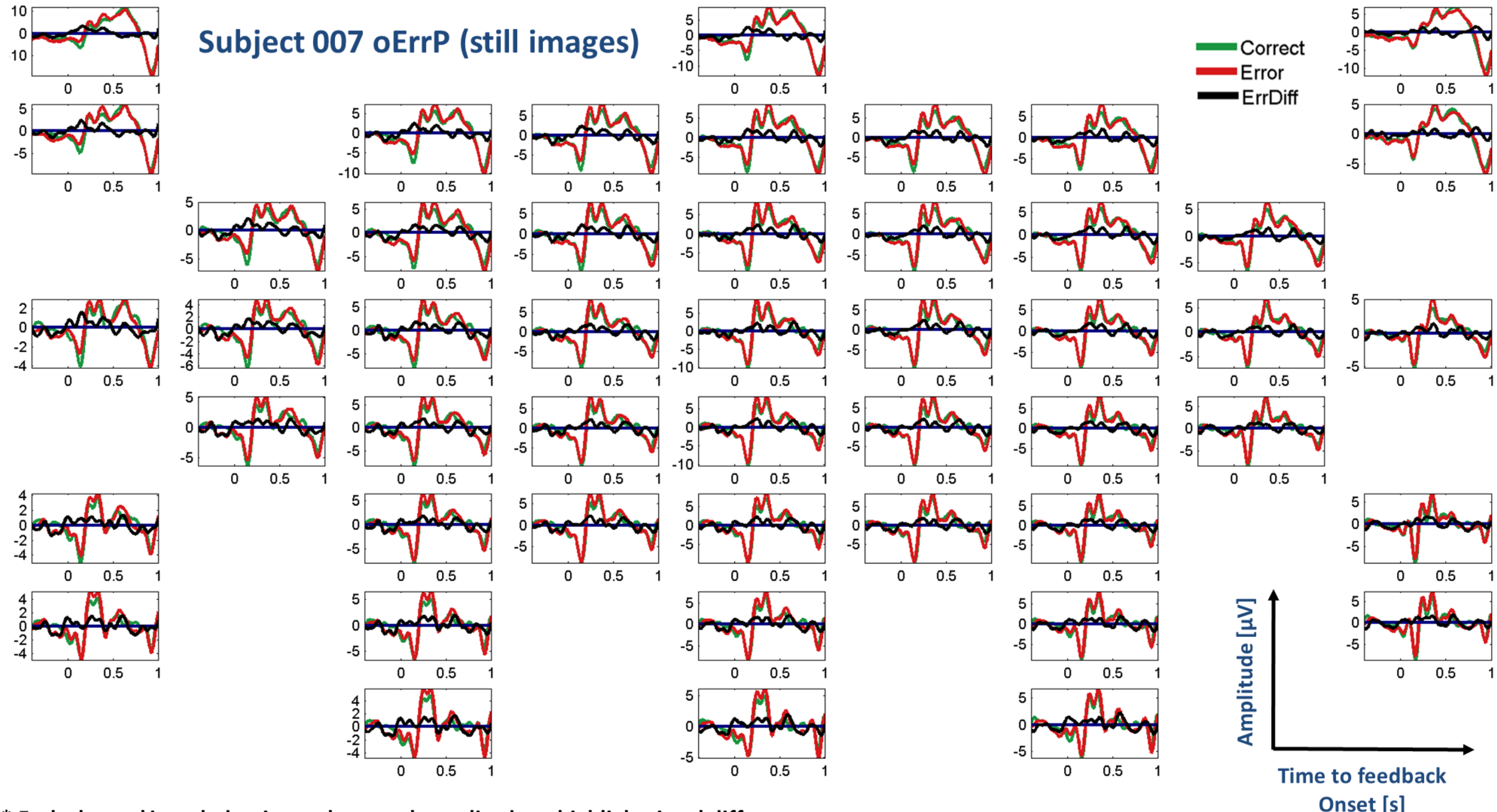
5.3 Average correct, incorrect, and ErrDiff traces (still images task oErrP).



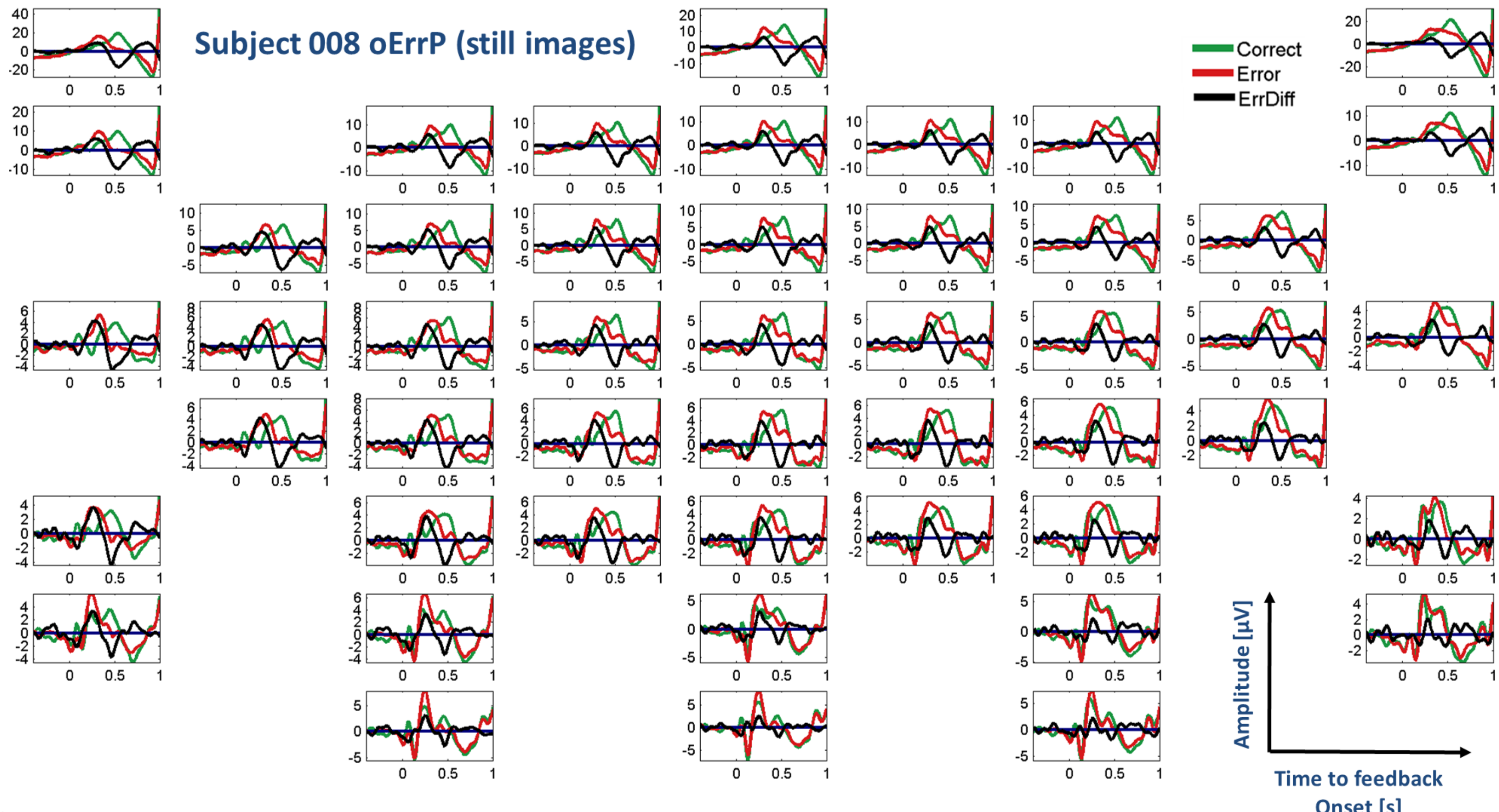
*** Each channel is scaled to its peak-to-peak amplitude to highlight signal differences**



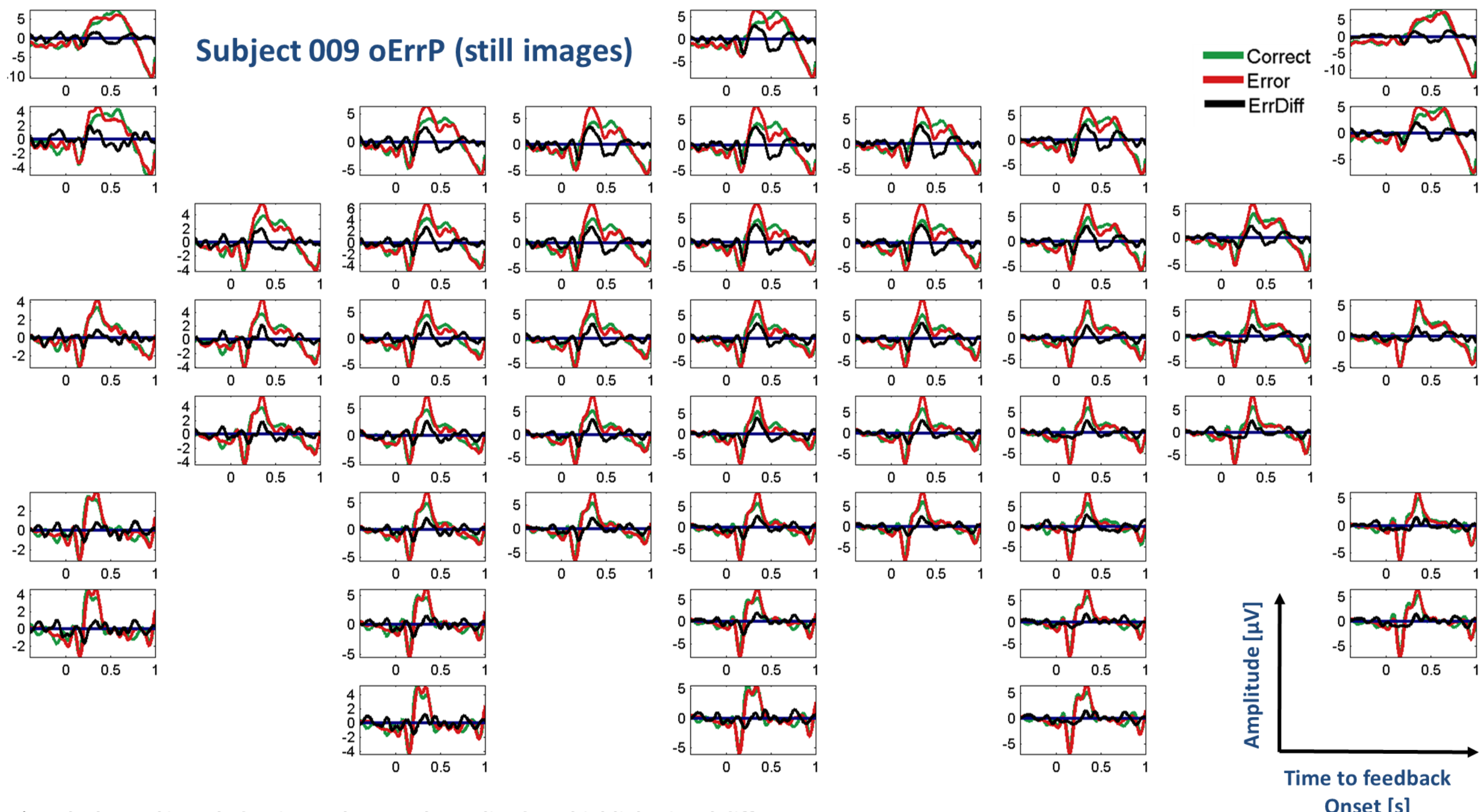
* Each channel is scaled to its peak-to-peak amplitude to highlight signal differences



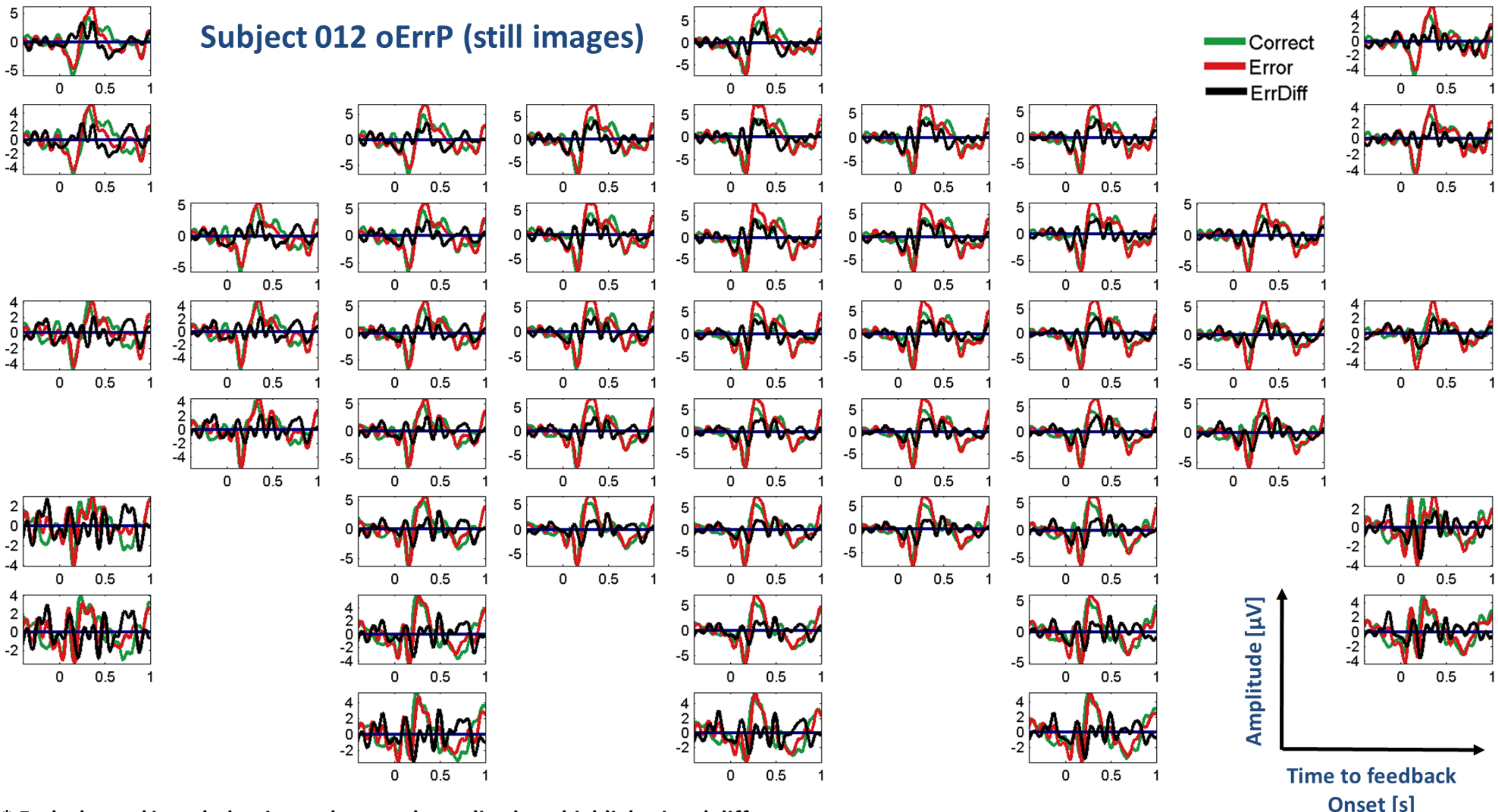
* Each channel is scaled to its peak-to-peak amplitude to highlight signal differences



* Each channel is scaled to its peak-to-peak amplitude to highlight signal differences

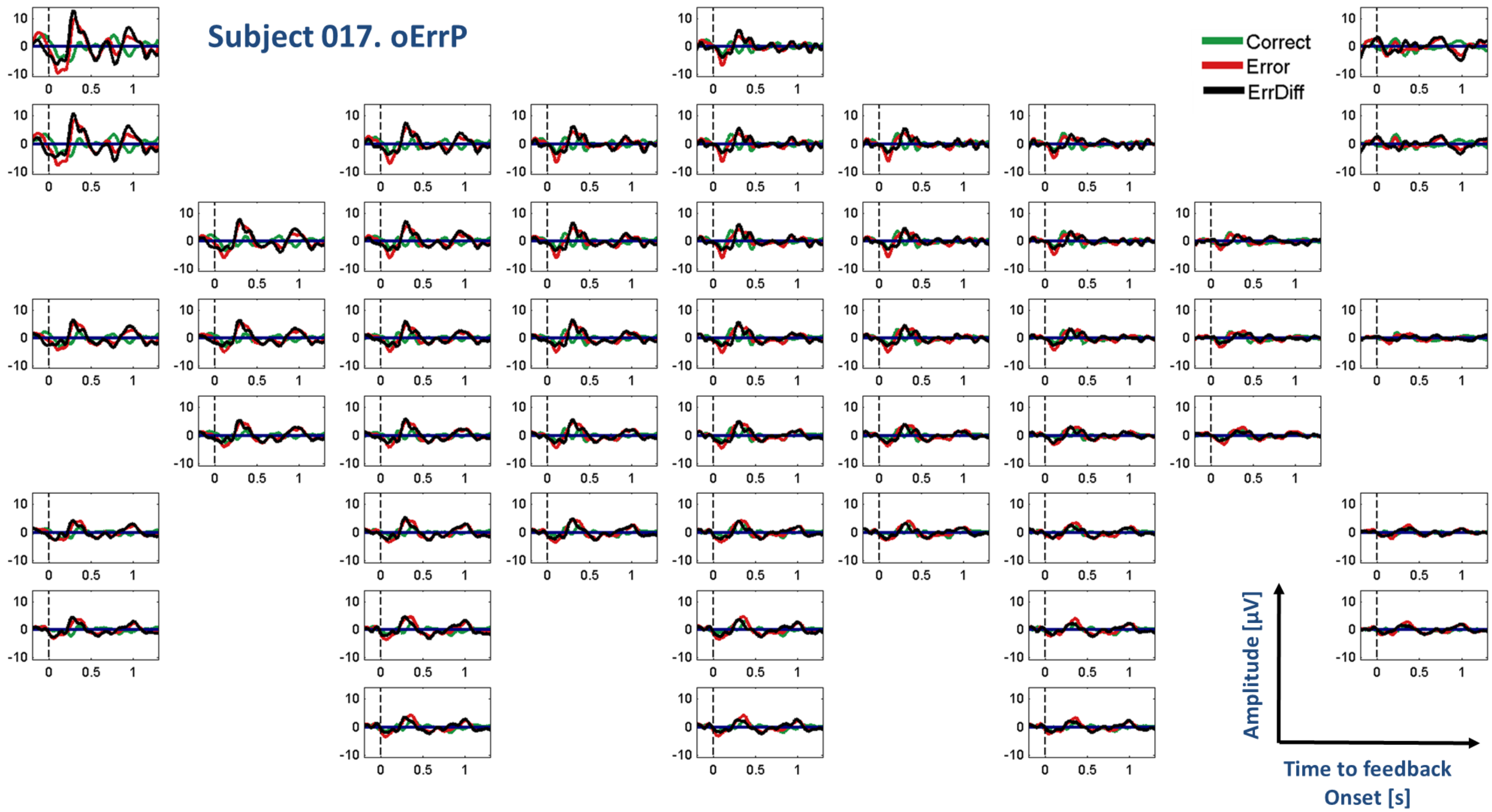


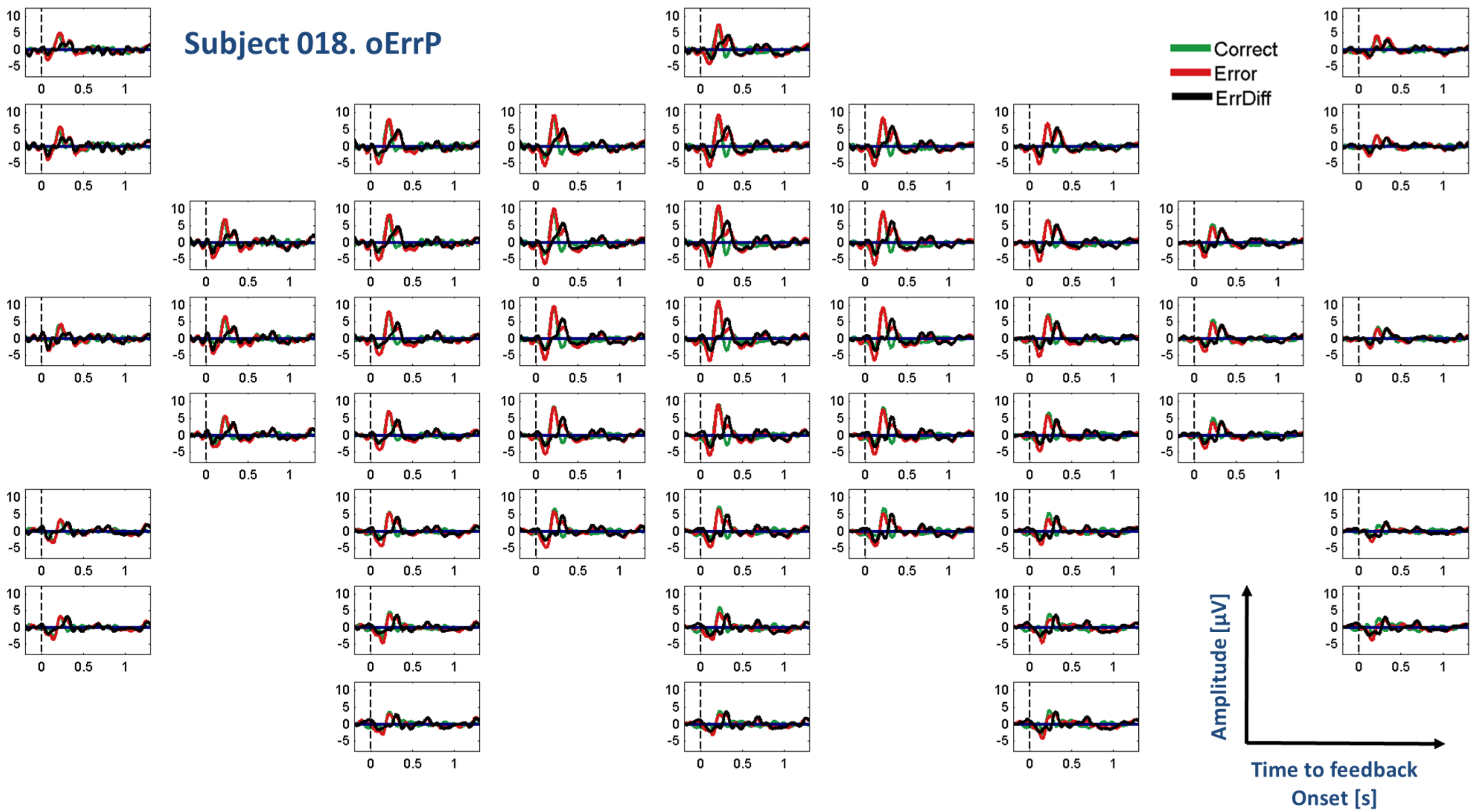
* Each channel is scaled to its peak-to-peak amplitude to highlight signal differences

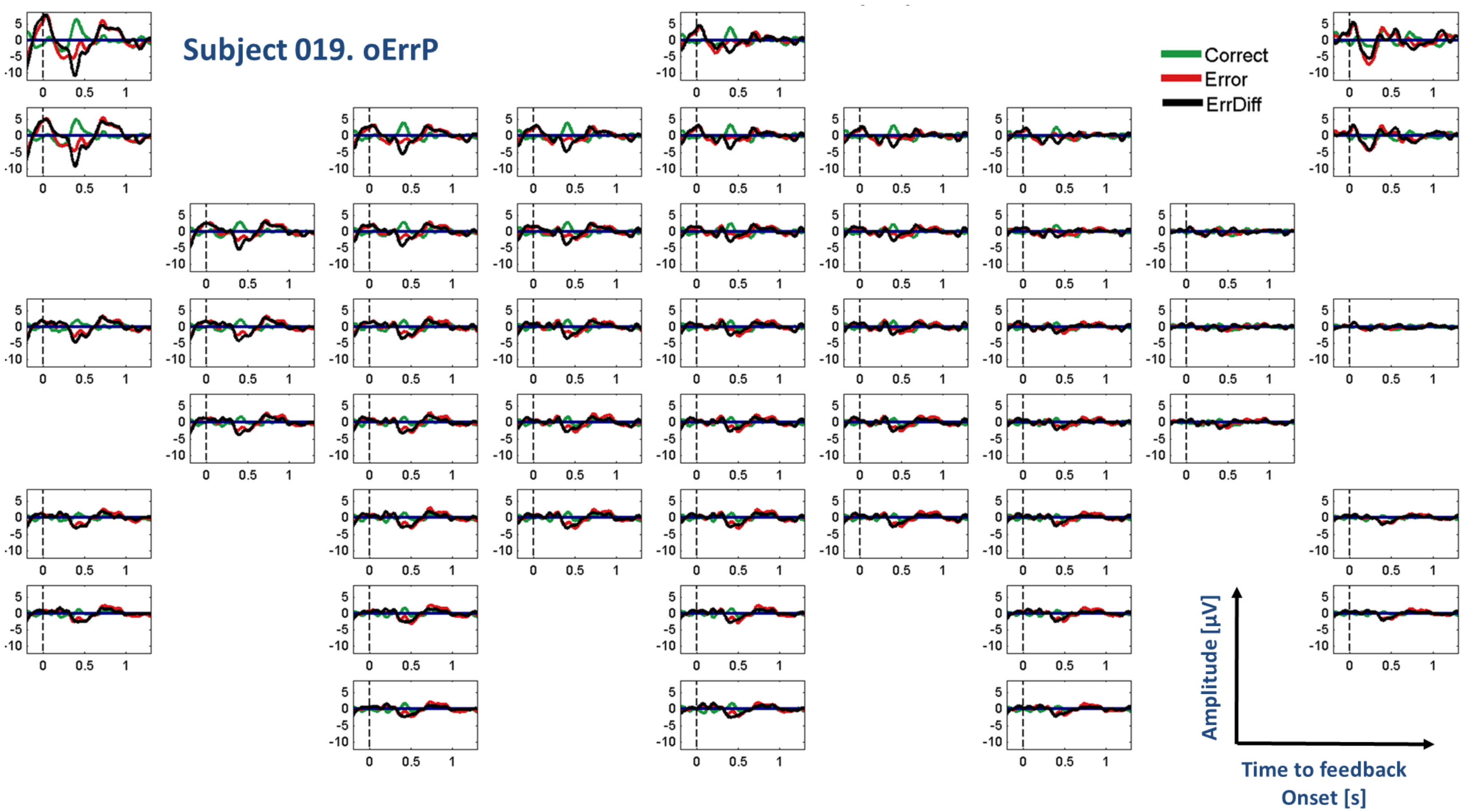


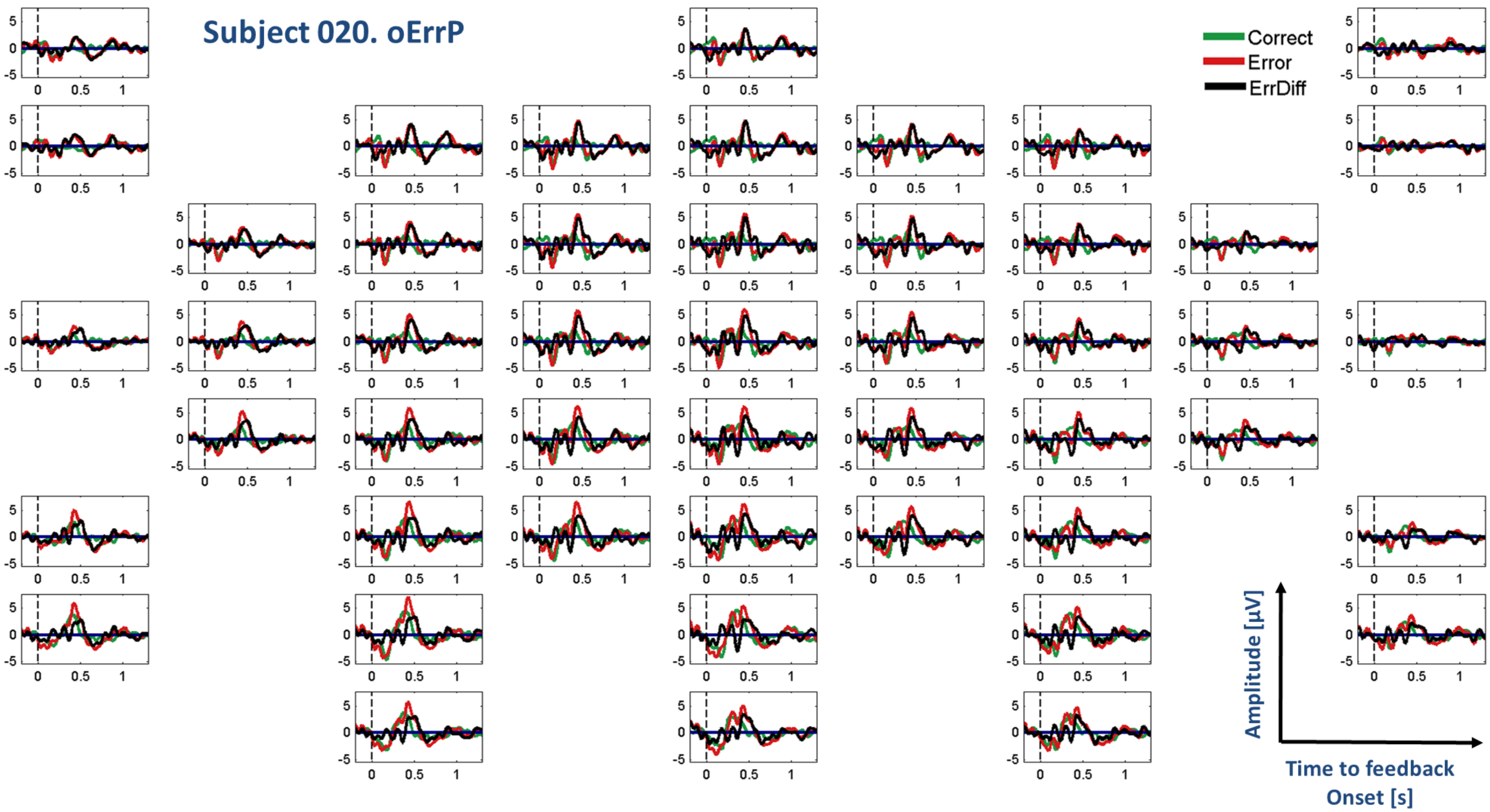
* Each channel is scaled to its peak-to-peak amplitude to highlight signal differences

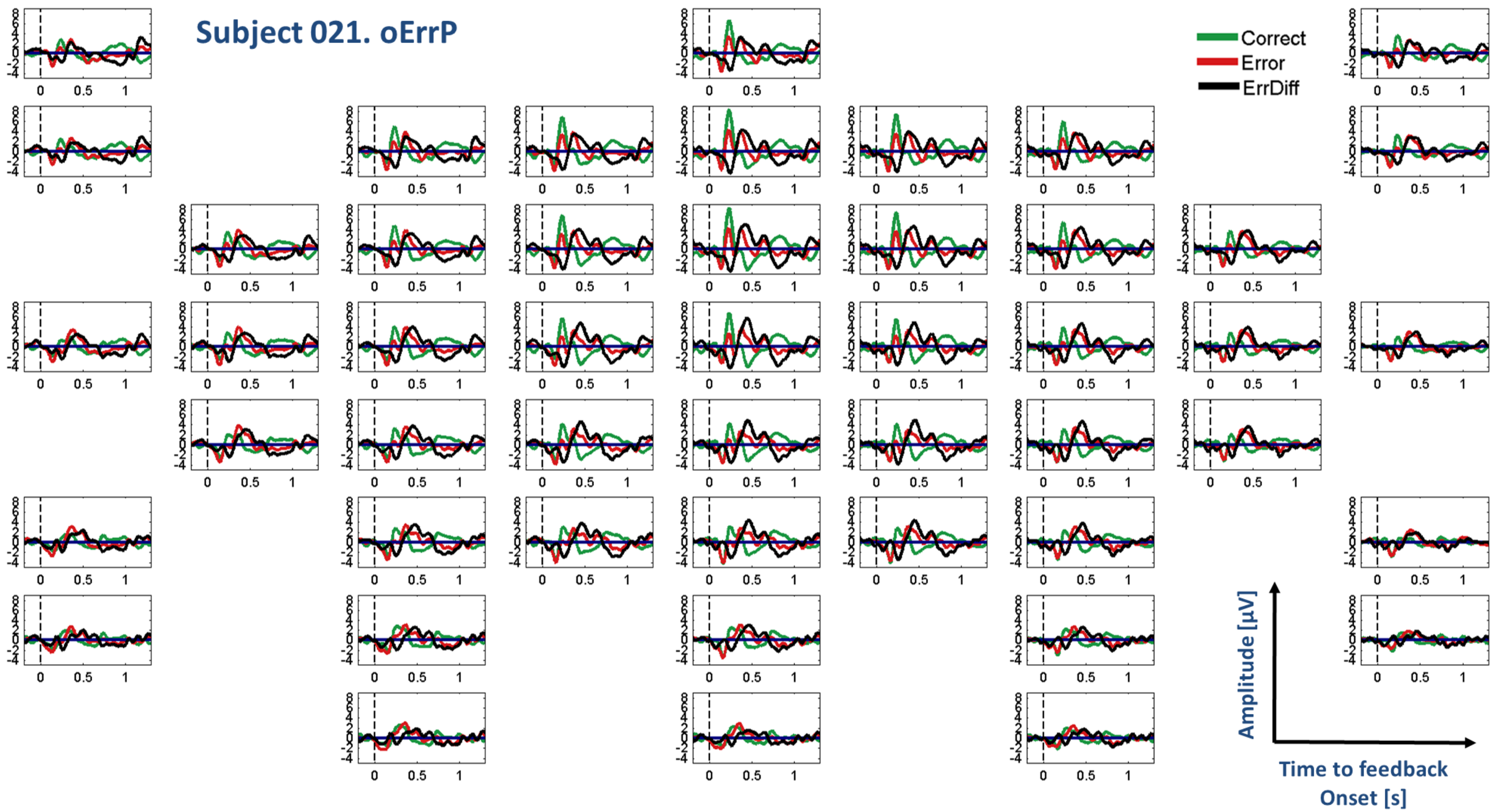
5.4 Average correct, incorrect, and ErrDiff traces (reaching task oErrP).



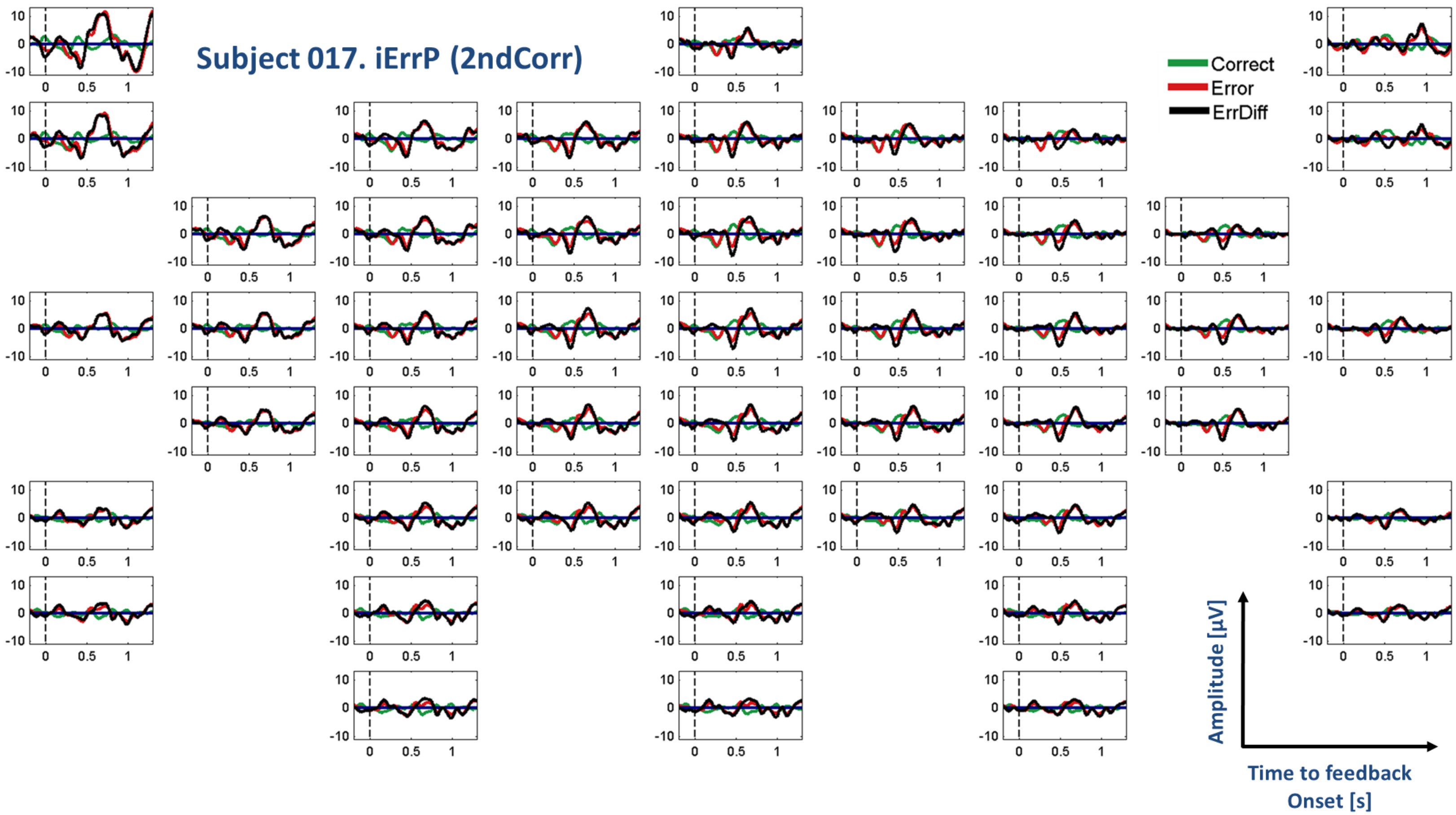

Subject 018. oErrP
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback Onset [s]

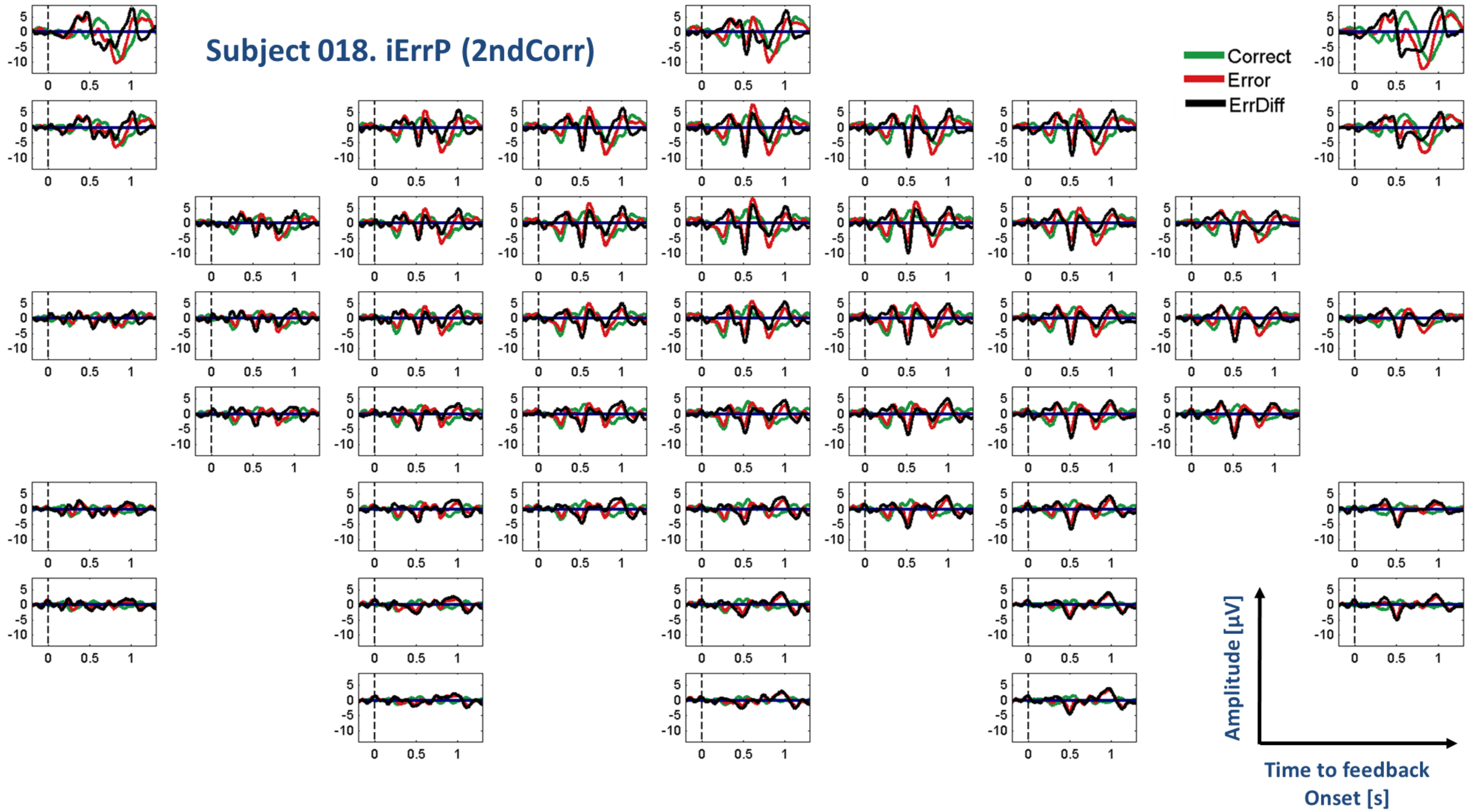

Subject 019. oErrP
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback Onset [s]


Subject 020. oErrP
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback Onset [s]

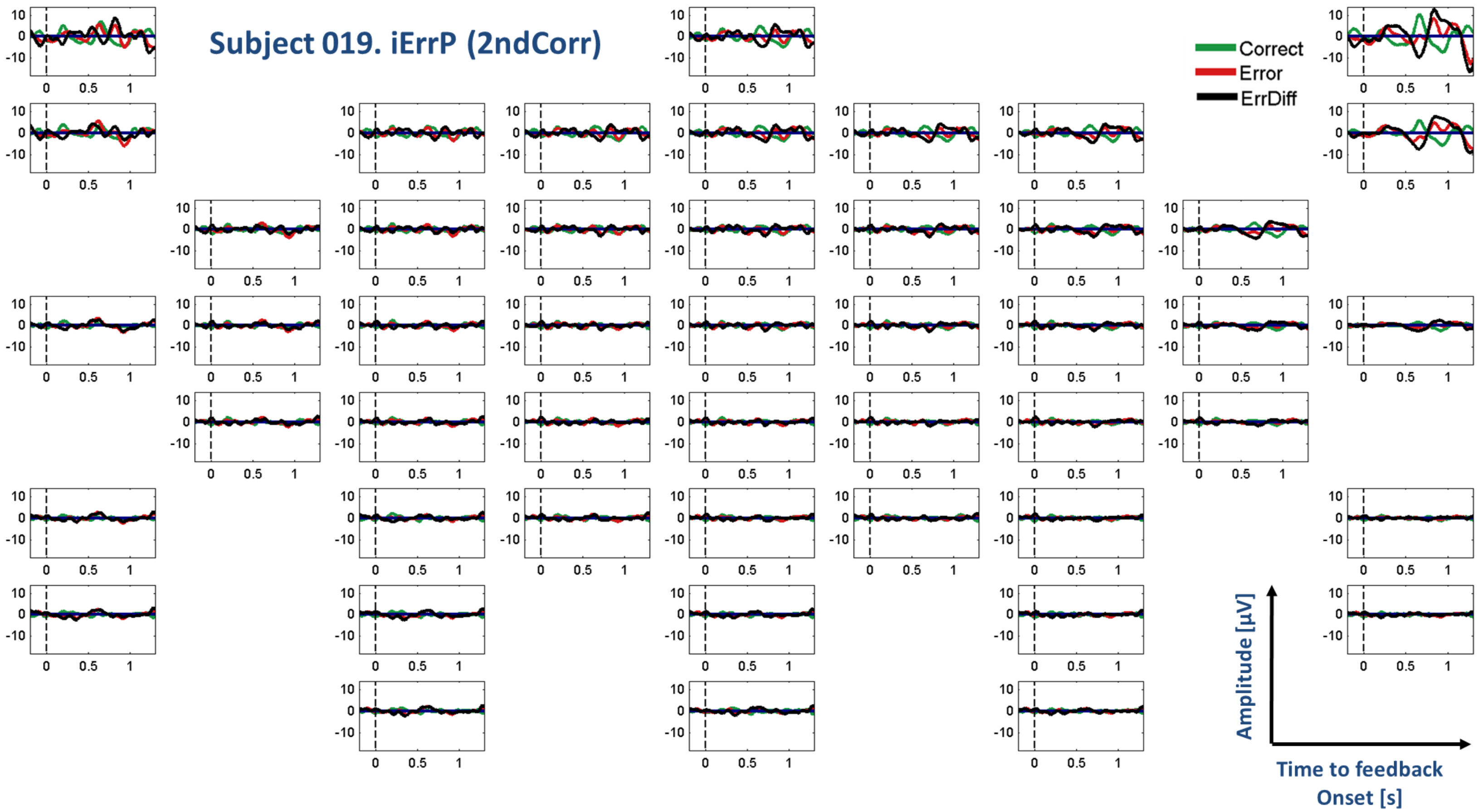

Subject 021. oErrP
Correct
Error
ErrDiff
Amplitude [μV]
Time to feedback Onset [s]

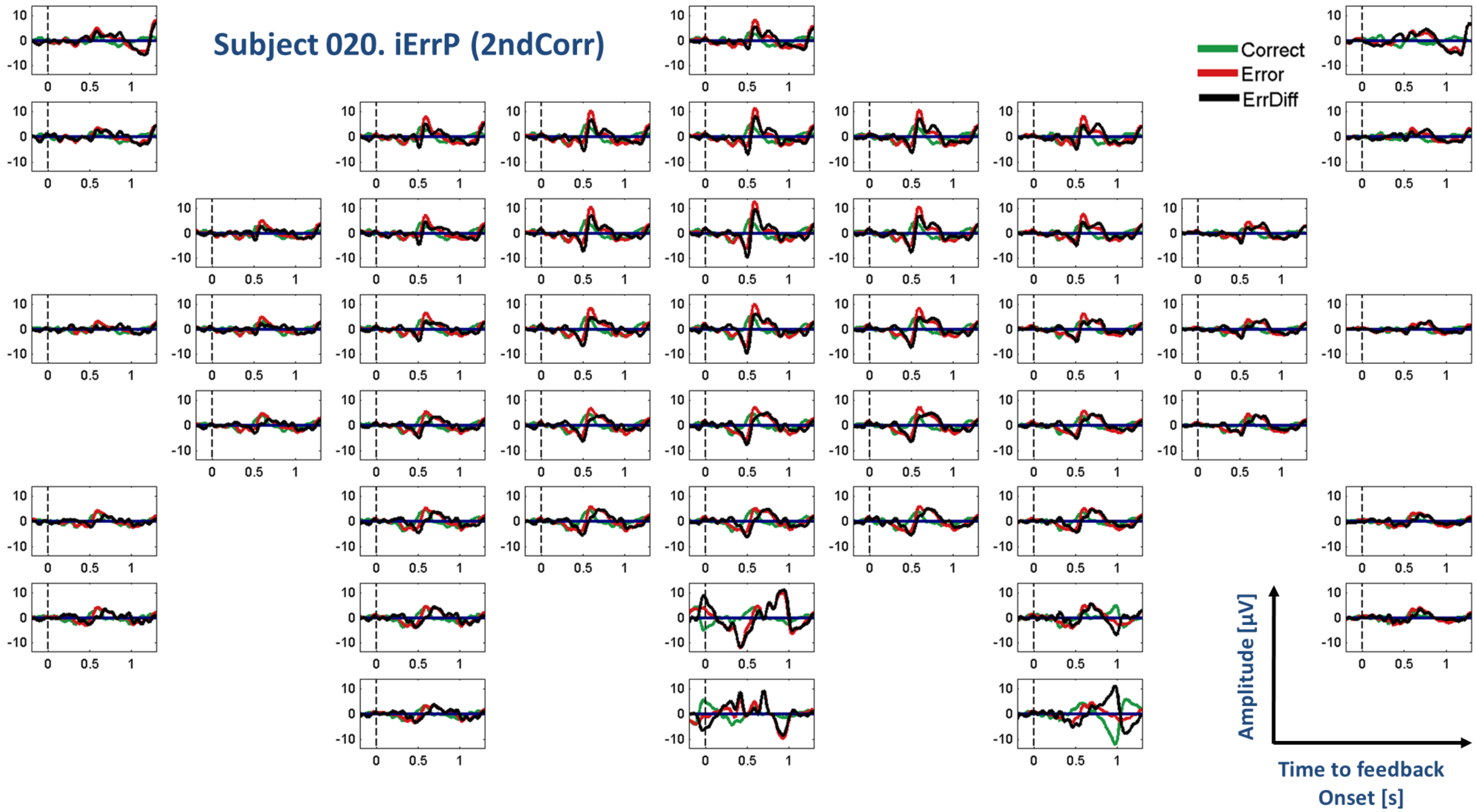
5.5 Average correct, incorrect, and ErrDiff traces (reaching task for iErrP 2ndCorr).

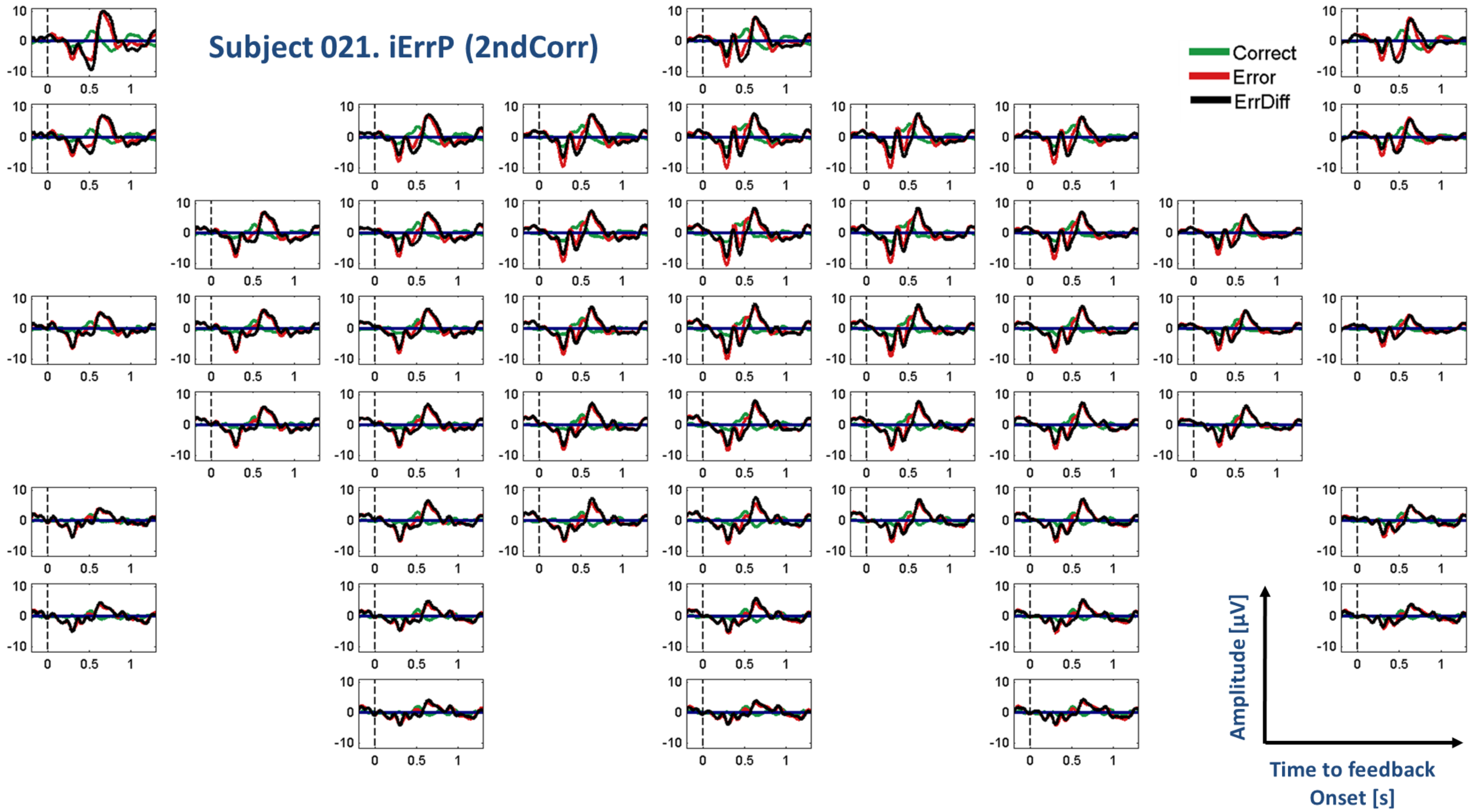




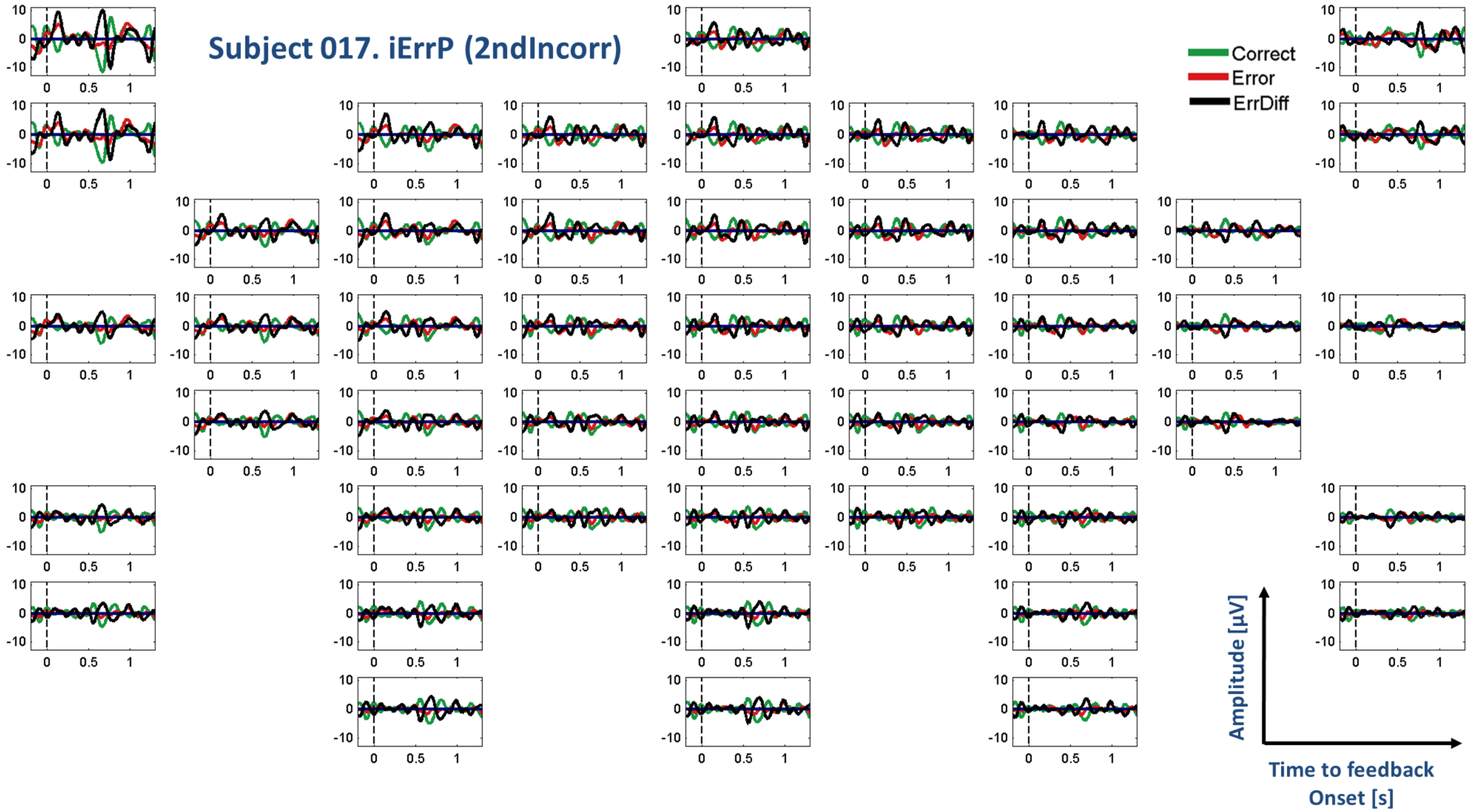
Subject 018. iErrP (2ndCorr)
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback Onset [s]

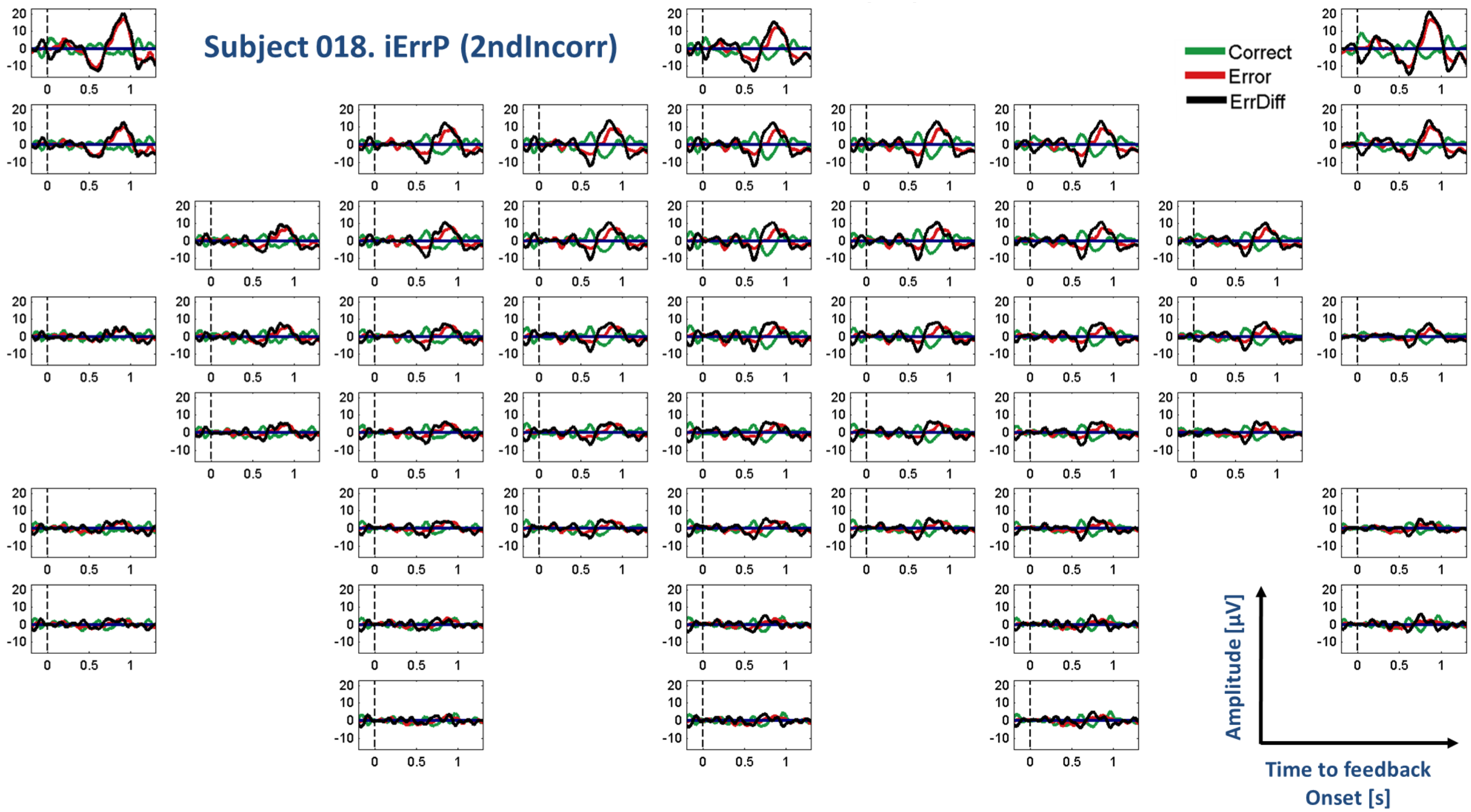

Subject 019. iErrP (2ndCorr)
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback Onset [s]

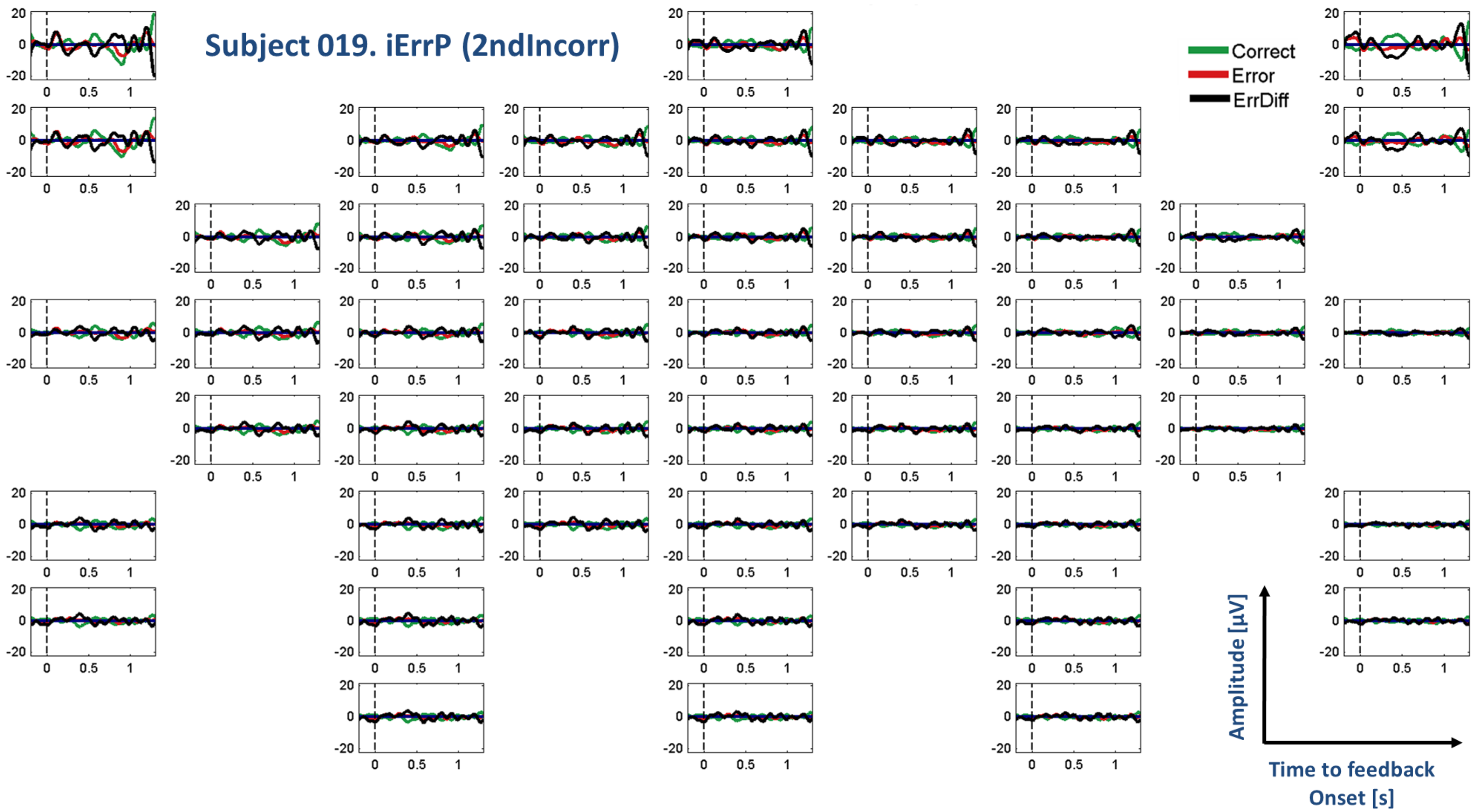

Subject 020. iErrP (2ndCorr)
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback
Onset [s]

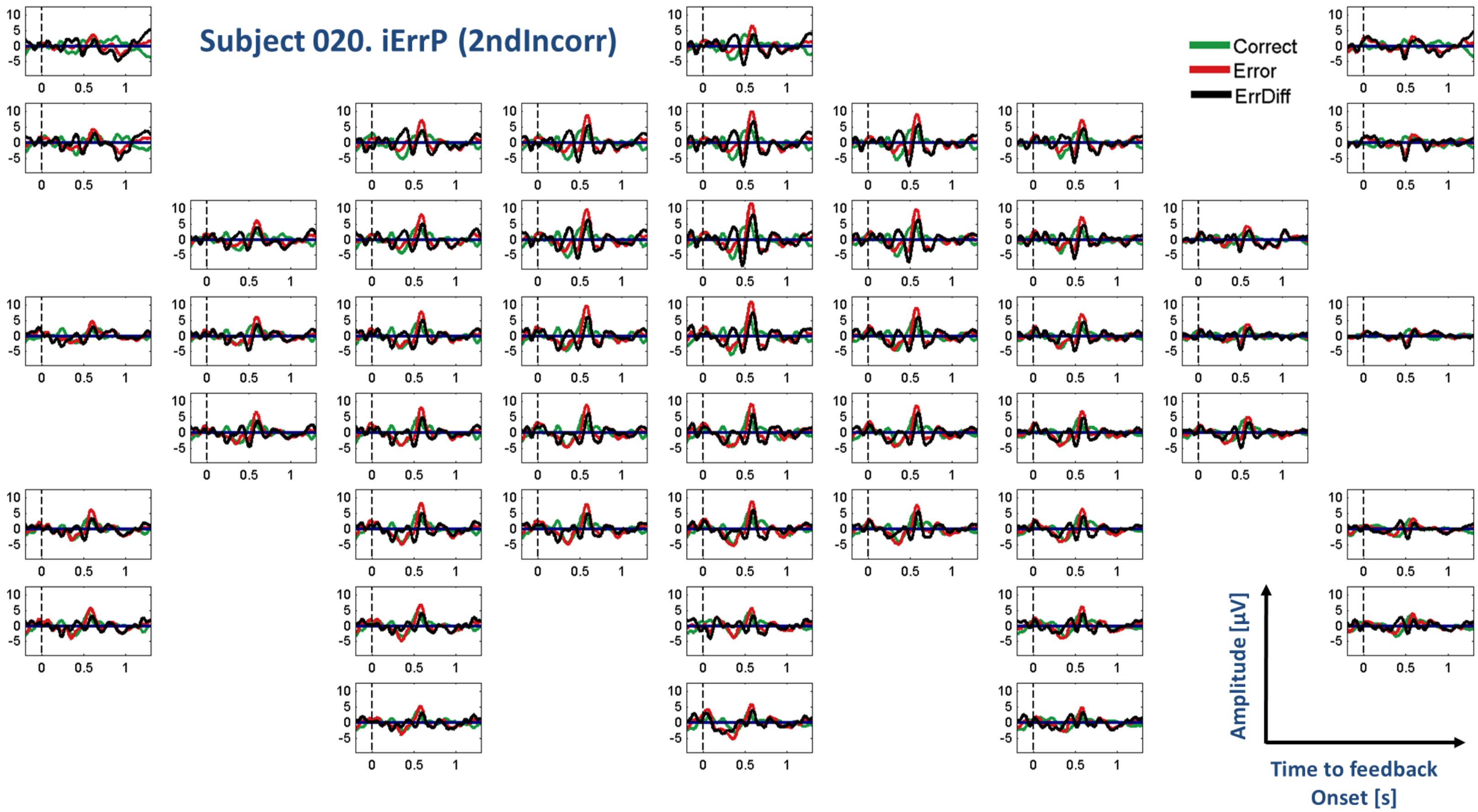

Subject 021. iErrP (2ndCorr)
Correct
Error
ErrDiff
Amplitude [μV]
Time to feedback Onset [s]

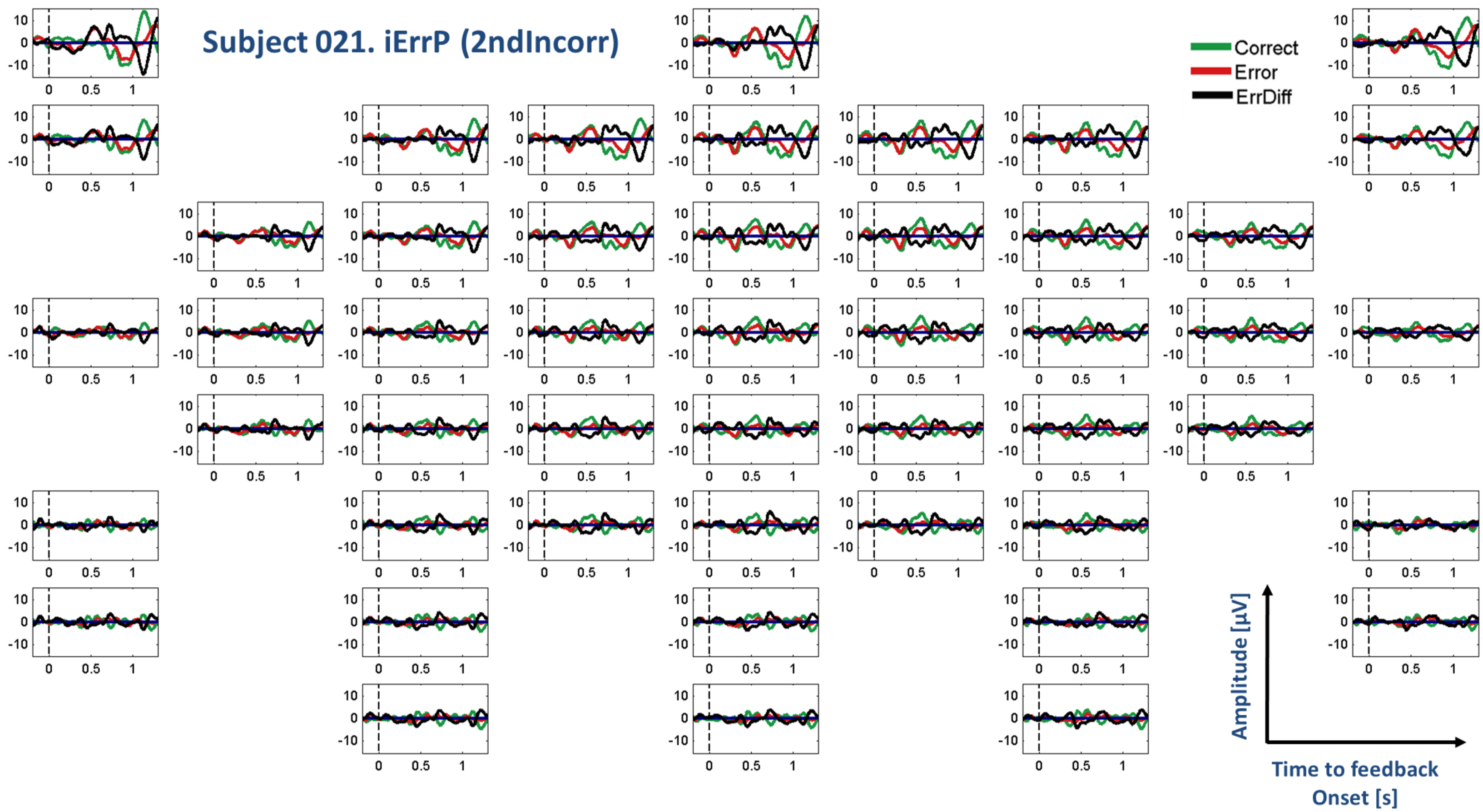
5.6 Average correct, incorrect, and ErrDiff traces (reaching task iErrP 2ndIncorr).




Subject 018. iErrP (2ndIncorr)
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback Onset [s]


Subject 019. iErrP (2ndIncorr)
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback Onset [s]


Subject 020. iErrP (2ndIncorr)
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback
Onset [s]


Subject 021. iErrP (2ndIncorr)
Correct
Error
ErrDiff
Amplitude [µV]
Time to feedback
Onset [s]

5.7 Secondary ErrP classification performance (offline)

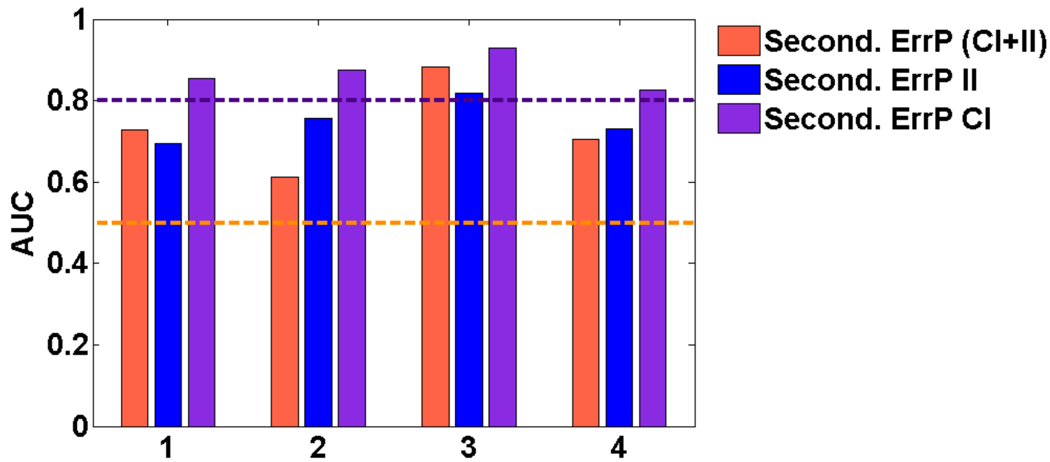


Figure III.25. Secondary ErrP classification performance for all naïve subjects (excluding the meditation subject).

Offline secondary ErrP (CI+II, II, CI) classification performance for all naïve subjects (except the meditation subject). Salmon, blue and violet bars depict offline classification values for secondary ErrP CI+II (2ndCorr and 2ndIncorr trials pooled together), secondary ErrP II (2ndIncorr) and secondary ErrP CI (2ndCorr) trials. X axis represents each one of the subjects.

5.8 Average correct, incorrect and ErrDiff traces from all naive subjects (iErrP 2ndIncorr).

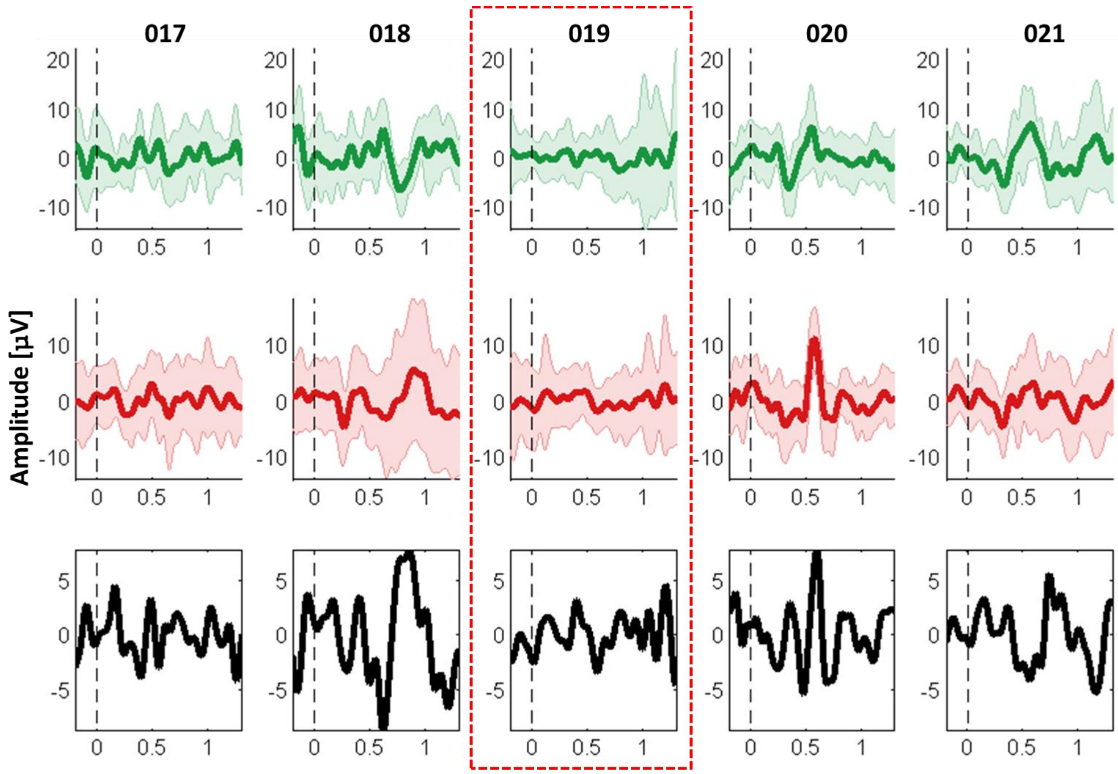


Figure III.26. Average correct, incorrect and ErrDiff traces from all naive subjects (iErrP 2ndIncorr).

Average traces for the ErrP decoder outcome class and their ErrDiff for all naïve subjects (after an incorrect selection by Baxter, 2ndIncorr). Columns represent the subject number; from left to right: 017, 018, 019, 020, 21.The correct (green), incorrect(red) and ErrDiff traces (iErrP) belong to electrode Cz. Error bars depict standard deviation of correct (light green) and incorrect (light red) trials. The X axis is the time to feedback onset (seconds), and the Y axis represents signal amplitude (µV). The red dashed-square presents subject M (019) traces. All traces from each column have the same scale. The amplitudes of the traces from subject 019 are similar to the ones from the other subjects.

CHAPTER IV: CONCLUSION

This dissertation evaluated the use of decoder error detection for adaptive decoding and volitional control. We focused on developing and implementing real-time single-trial systems for decoding error-related potentials from invasive and non-invasive recordings. Beyond proof of concept for successful single-trial and closed-loop ErrP detection, we were able to compare the error-related potentials from different tasks, signal modalities, and species. Here we summarize the contributions that this thesis makes to the fields of BMI, performance monitoring and human-robot interaction. We also discuss future directions of research with an emphasis on the use of ErrP for augmentative and alternative communication (AAC).

1. Innovation and Impact

1.1 Contributions to performance monitoring during BMI control

Error-related potentials have been thoroughly explored and described under non-invasive BMI control (Ferrez & Millan, 2005; Chavarriaga & Millan, 2010; Schalk et al., 2000). The same has been done for error-related activity during non-BMI tasks using invasive and non-invasive recordings (Emeric et al., 2008; Emeric et al., 2010; Godlove et al., 2011; Botvinick et al., 2004; Klein et al., 2007; Falkenstein et al., 2000; Gehring et al., 1993; Amiez et al., 2005). Nevertheless, the study of error-related activity under BMI control using invasive technologies is nascent (Geng et al., 2013; Prins et al., 2013; Milekovic et al., 2012). In Chapter II we characterized error-related local field potentials (eLFP) observed during a BMI task, specifically while a non-human primate (NHP) controlled a saccade BMI, bridging current knowledge of error-related potentials under

BMI and non-BMI control scenarios. To our knowledge, this is the first real-time LFP-based decoder error detection system integrated to an invasive BMI.

Our results suggest that the ErrP signal and its associated ErrDiff traces present traits similar to the ones previously described in the literature under BMI and non-BMI control (in invasive and non-invasive recordings) both in time and frequency. Averaging the correct and incorrect trials of a population of sessions from each monkey, and focusing on their ErrDiff traces, we found a characteristic small negative trough (100ms) and a large positive peak (~200ms) in the SEF array of both monkeys. These results are consistent with observations made in eLFP from the SEF of NHP during non-BMI go-no go tasks (Emeric et al., 2010), as well as in EEG ErrP from humans during BMI control (Ferrez & Millan, 2005; Chavarriaga et al., 2014). We also found that the ErrDiff traces presented information about the magnitude and valence of the decoder errors. The level of discrepancy (or conflict) significantly modulated the amplitude of the positive peaks of the eLFP in the SEF arrays of both monkeys, a result previously reported by Emeric et al. (2010) in SEF during a non-BMI task.

From the time-frequency content, we found, in the ErrDiff spectrograms of both monkeys, increases in delta, theta, and alpha activity (200-400ms after the feedback onset) and a strong decrease in beta power (200-800ms after the feedback onset). The increases in delta and theta power during error processing have been documented in non-BMI tasks (Yordanova et al., 2004; Cavanagh et al., 2010; Cohen, 2011) while the increases in alpha and the decreases in beta have been previously reported in ErrP during BMI control (Chavarriaga et al., 2014). The decreases in beta seem to be related to a sense of agency during the BMI task, and were not observed when the subjects merely monitored errors performed by external agents.

Furthermore, we found increases and decreases in the high-gamma band of all the SEF and FEF channels in both monkeys. That is, a decrease of activity lasting 200ms in both monkeys (between 200ms and 300ms post feedback onset), immediately followed by an increase in high-gamma with duration of 200ms. Increases in high gamma during error production have been reported on ECoG data from humans during a motor task (Milekovic et al., 2012) but last longer (~200-500ms) and do not show prior decreases in activity in the same frequency band. Increases of high frequency power in LFP during decoder errors have been reported during BMI control (250ms post-feedback onset), but only in preliminary results in marmoset monkeys (Geng et al., 2013). The significance of these changes in frequency is not clear, but the similarity of the results across different recording modalities (LFP, ECoG, EEG), different tasks (BMI and non-BMI control), and species (humans and NHP) suggest a common source for the signals.

1.2 ErrP-based adaptive decoding as a standard feature in BMI design

The use of decoder error detection for adaptive decoding has become a common practice in non-invasive BMI (Chavarriaga et al., 2014; Millán et al., 2010) but a rarely used one in invasive BMI systems. In Chapter II we demonstrated that decoder errors can be detected from intracranial electrodes on a single-trial basis with high accuracy in a reliable and fast way. We also demonstrated that the characteristics of the error-related potentials are very similar across recording modalities (LFP, ECoG, and EEG) and tasks (under BMI and non-BMI control) suggesting a unifying approach could be taken for detecting eLFP. These results suggest that error-based adaptive decoding can become a standard feature in BMI design.

1.3 Advantages of ErrP for direct volitional control

BMI researchers are constantly in search of better signal processing and decoding algorithms that can boost decoding performance in both invasive and non-invasive BMI (Millán et al., 2010; Baranauskas, 2014). Although several streams of research are pursuing this purpose, less attention has been given to the paradigms implemented and the constraints these impose on the users. Learning a new mental task, constantly using external stimuli, increasing the levels of attention and cognitive processing are common to most BMI. Nevertheless, for long-term BMI control or for its use by locked-in people, these paradigm constraints negatively affect BMI performance (Kübler & Birbaumer, 2008; Schnakers et al., 2008). Error-related potentials are naturally occurring potentials that do not need to be learned, nor do they require high levels of attention or cognitive processing; they are observed in multiple tasks and found in the majority of the population (Ullsperger et al., 2014; Chavarriaga et al., 2014; Iturrate et al., 2014). In Chapter III the use of ErrP for direct volitional control was presented as an alternative to current paradigms by employing the ErrP signals as an indicator of the user's expectations under a binary-choice task. Our results demonstrated that ErrP can be used for direct control via binary selection and, given the appropriate levels of task engagement and agency, could enable communication between the user and any external agent. Our implementation, although technically more challenging, proved that single-trial real-time closed-loop decoding of ErrP is possible, and that decoding performance can be improved if the closed-loop system performs decoding of both the observation and interaction ErrP. For the locked-in community specifically, such a system would foster the goal of real-time intuitive human-robot interaction and brain-machine control.

1.4 Contributions to the study of the effect of meditation on attention

Although preliminary (only from subject M), our results suggest ErrP could be used as a marker of attentional changes derived from the practice of meditation. In Chapter III we hypothesized that subjects participating in the reaching task in a meditative state would be more focused and less prone to be affected by the noise and distractions from the surrounding environment. Our results proved otherwise: the error-related signals were smaller in amplitude and the subject seemed less engaged in the task. Average correct, average incorrect and the ErrDiff traces from oErrP showed smaller peak amplitudes. Average incorrect and ErrDiff traces from the iErrP (2ndCorr) also presented smaller peak amplitudes, especially the average incorrect trace. It is thought that error-related potentials signal a boost in attention to the incorrect actions (O'Connell et al., 2007; Vocat et al., 2008) and that the valence and magnitude of the error is encoded in the ErrP (Iturrate et al., 2010; Emeric et al., 2010). For these reasons it is possible that while a person is in a meditative state, the low peak amplitudes of the ErrDiff traces reflect decreases in the boost of attention that the errors should trigger. Taking these arguments into consideration, ErrP signals could be employed to study how attention and the arousal levels are modified by meditation.

2. Future Directions

2.1 Error-related LFP and spike activity during BMI control

Our analysis of error-related activity during BMI control only covered the activity of LFP from the SEF, FEF and PFC. Concurrent analyses of spike and LFP data must be performed to better understand the neural correlates of error processing and performance monitoring in these areas, and others involved in goal-oriented behavior. Of particular interest is the role of these signals in adaptation and learning. Understanding adaptation and learning during BMI control could lead to the development of better and more robust BMI technologies by taking into account the intrinsic non-stationarities added by these cognitive processes. Future work should focus on studying such dynamics, the cognitive rhythms coded in the error-related LFP and spike activity, and the possible cross-frequency amplitude-amplitude and phase-amplitude coupling between recording sites.

2.2 Effect of meditation on attention studied via ErrP

Based on our preliminary results, future work should explore in depth the effect meditation has on error-related potentials and their connection to attention and arousal. A binary-choice task with high levels of engagement (aiming to evoke iErrP) could be implemented with subjects participating in the task before, during, and after meditation. The study should focus on iErrP since these signals have stronger responses than the oErrP, possibly making the analysis of the evoked-potentials easier. The subjects can be informed that the binary-choice task is directly controlled by their EEG activity (even if not) to increase their engagement and to provide a sense of agency. The results of this study could shed light on the effect meditation has on attention, performance monitoring and error processing.

2.3 ErrP for augmentative and alternative communication (AAC)

With the advent of new communication modalities, the use of binary signals for communication seems more plausible (Ahani et al., 2014; Bacher et al., 2015; Millán et al., 2010). The use of ErrP for direct volitional control was explored with the idea of providing the locked-in population a means of interaction with the world. Of particular interest is the possibility of employing ErrP for communication. The results provided in Chapter III suggest that ErrP could be used for selecting letters or icons via binary classification. Although time consuming and prone to errors, if our ErrP decoder presents a correct selection probability of over 70% (Perelmouter & Birbaumer, 2000; Birbaumer et al., 1999) we could provide successful spelling capabilities using different communication interfaces (Ahani et al., 2014; Higger et al., 2014).

In the case of the Bayesian framework developed by Higger et al. (2014), spelling is accomplished using a Bayesian probabilistic model of accurate choice detection. The requirements are setting the framework to binary selection, providing prior knowledge of the ErrP detection performance (confidence interval), and choosing a minimum desired spelling accuracy. If our ErrP detection accuracy is high, the Bayesian framework will require fewer attempts to properly select a letter under the expected spelling accuracy. The lower the ErrP decoding performance value, the more observations (trials) the spelling framework will require to properly select a letter.

Although time consuming, this process has been proven with SSVEP spelling. Future work should focus on implementing ErrP detection for communication.

LIST OF JOURNAL ABBREVIATIONS

Arch Phys Med Rehabil	Archives of Physical Medicine and Rehabilitation
Conf Proc EUSIPCO	Proceeding of the European Signal Processing Conference
Conf Proc IEEE/EMBS	Proceedings of the International IEEE Conference on Engineering in Medicine and Biology Society
Conf Proc IEEE/ROBIO	Proceedings of the International IEEE Conference on Robotics and Biomimetics
Conf Proc IEEE Robots Automat	Proceedings of the International IEEE Conference on Robotics and Automation
Conf Proc IJCAI	Proceedings of the International Joint Conference on Artificial Intelligence
Electroencep. Clin. Neurophysiol	Electroencephalography and Clinical Neurophysiology
ESANN	Proceedings of the European Symposium on Artificial Neural Networks
Expert. Rev. Med. Devices	Expert Review of Medical Devices
ICCN	Proceedings of the International Conference on Cognitive Neurodynamics
IEEE Trans Biomed Eng	IEEE Transactions on Biomedical Engineering
IEEE Trans Neural Rehab Sys Eng	IEEE Transactions on Neural and Rehabilitation Systems Engineering

IEEE Trans Rehab Eng	IEEE Transactions on Rehabilitation Engineering
IEEE Trans Robotics	IEEE Transaction on Robotics
J Cogn Neurosc	Journal of Cognitive Neuroscience
J Exp Psych	Journal of Experimental Psychology: Human Perception and Performance
J Head Trauma Rehabil	Journal of Head Trauma Rehabilitation
J Neural Engineering	Journal of Neural Engineering
J Neurol Neurosurg Psychiatry	Journal of Neurology, Neurosurgery, and Psychiatry
J Neurology	Journal of Neurology
J Neurophysiol	Journal of Neurophysiology
J Neurosc	Journal of Neuroscience
J Neurosc Methods	Journal of Neuroscience Methods
Med Biol Eng Comput	Medical & Biological Engineering & Computing
Neurorehabil and Neural Repair	Neurorehabilitation and Neural Repair
PNAS	Proceedings of the National Academy of Sciences of the USA
Psychol. Science	Psychological Science
Speech Commun	Speech Communications
Trends in Cogn Sci	Trends in Cognitive Sciences

BIBLIOGRAPHY

(ACRM), A.C. of R.M., 1995. Recommendations for use of uniform nomenclature pertinent to patients with severe alterations in consciousness. *Arch Phys Med Rehabil*, 76(February), pp.205–209.

(AHA), A.H.A., 2016. Heart disease, stroke and research statistics at a glance. *American Heart Association (AHA) Website*, (1), pp.1–5. Available at: http://www.heart.org/idc/groups/ahamah-public/@wcm/@sop/@smd/documents/downloadable/ucm_480086.pdf.

Acqualagna, L. et al., 2010. A novel brain-computer interface based on the rapid serial visual presentation paradigm. In *Conf Proc IEEE/EMBS*. Buenos Aires, Argentina, pp. 2686–2689.

Ahani, A. et al., 2014. RSVP IconMessenger : icon-based brain-interfaced alternative and augmentative communication. *Brain-Computer Interfaces*, 1, pp.192–203.

Alexander, W.H. & Brown, J.W., 2011. Medial prefrontal cortex as an action-outcome predictor. *Nature neuroscience*, 14(10), pp.1338–44.

Allison, B.Z. & Neuper, C., 2010. Brain-Computer Interfaces. In D. S. Tan & A. Nijholt, eds. *Brain-computer interfaces: Applying our Minds to Human-Computer Interaction*. Springer Berlin Heidelberg, pp. 35–54.

Amiez, C., Joseph, J.-P. & Procyk, E., 2005. Anterior cingulate error-related activity is modulated by predicted reward. *European Journal of Neuroscience*, 21, pp.3447–3452.

Bacher, D. et al., 2015. Neural Point-and-Click Communication by a Person With Incomplete Locked-In Syndrome. *Neurorehabilitation and Neural Repair*, 29(5), pp.462–471.

Bansal, A.K. et al., 2012. Decoding 3D reach and grasp from hybrid signals in motor and premotor cortices: spikes, multiunit activity, and local field potentials. *J Neurophysiol*, 107(5), pp.1337–1355.

Barachant, A. et al., 2012. BCI Signal Classification using a Riemannian-based kernel. In *20th ESANN*. Bruges, Belgium, pp. 97–102.

Barachant, A. et al., 2013. Classification of covariance matrices using a Riemannian-based kernel for BCI applications. *Neurocomputing*, 112, pp.172–178.

Barachant, A., 2014. *MEG decoding using Riemannian Geometry and Unsupervised classification. Notes on the winner of the Kaggle "DecMeg2014 - Decoding the*

Human Brain" competition., https://www.kaggle.com/c/decoding-the-human-brain.

Barachant, A. & Bonnet, S., 2011. Channel selection procedure using riemannian distance for BCI applications. In *Conf Proc IEEE/EMBS on Neural Engineering*. Cancun, Mexico, pp. 348–351.

Barachant, A. & Congedo, M., 2014. A plug & play P300 BCI using information geometry. *arXiv preprint arXiv:1409.0107*, pp.1–9. Available at: http://arxiv.org/abs/1409.0107.

Baranauskas, G., 2014. What limits the performance of current invasive brain machine interfaces? *Frontiers in Systems Neuroscience*, 8(April), p.68.

Bauer, G., Gerstenbrand, F. & Rumpl, E., 1979. Varieties of the locked-in syndrome. *J Neurology*, 221(2), pp.77–91.

Beaudoin, N. & De Serres, L., 2010. Locked-in Syndrome J. Stone & M. Blouin, eds. *International Encyclopedia of Rehabilitation*. Available at: http://cirrie.buffalo.edu/encyclopedia/en/article/303/.

Bhattacharyya, S., Konar, A. & Tibarewala, D.N., 2014. Motor imagery, P300 and error-related EEG-based robot arm movement control for rehabilitation purpose. *Med Biol Eng Comput*, 52(12), pp.1007–1017.

Birbaumer, N. et al., 1999. A spelling device for the paralysed. *Nature*, 398(6725), pp.297–298.

Birbaumer, N., 2006a. Brain-computer-interface research: coming of age. *Clinical Neurophysiology*, 117(3), pp.479–483.

Birbaumer, N., 2006b. Breaking the silence: Brain-computer interfaces (BCI) for communication and motor control. *Psychophysiology*, 43(6), pp.517–532.

Blain-Moraes, S. et al., 2012. Barriers to and mediators of brain-computer interface user acceptance: focus group findings. *Ergonomics*, 55(5), pp.516–525.

Blumberg, J. et al., 2007. Adaptive classification for brain computer interfaces. In *Conf Proc IEEE /EMBS*. Cité Internationale, Lyon, France, pp. 2536–2539.

Bokil, H. et al., 2010. Chronux: A platform for analyzing neural signals. *J Neurosc Methods*, 192(1), pp.146–151.

Botvinick, M.M., Cohen, J.D. & Carter, C.S., 2004. Conflict monitoring and anterior cingulate cortex: an update. *Trends in Cognitive Sciences*, 8(12), pp.539–546.

Brainard, D.H., 1997. The Psychophysics Toolbox. *Spatial Vision*, 10, pp.433–436.

Brincat, S.L. & Miller, E.K., 2015. Frequency-specific hippocampal-prefrontal interactions during associative learning. *Nature Neuroscience*, 18(4), pp.576–581.

Brumberg, J.S., Nieto-Castanon, A., et al., 2011. Brain-Computer Interfaces for Speech Communication. *Speech Commun*, 52(4), pp.367–379.

Brumberg, J.S., Wright, E.J., et al., 2011. Classification of intended phoneme production from chronic intracortical microelectrode recordings in speech-motor cortex. *Frontiers in Neuroscience*, 5(May), p.12.

Brumberg, J.S. & Guenther, F.H., 2010. Development of speech prostheses: current status and recent advances. *Expert. Rev. Med. Devices*, 7(5), pp.667–679.

Bruno, M.-A. et al., 2011. A survey on self-assessed well-being in a cohort of chronic locked-in syndrome patients: happy majority, miserable minority. *BMJ Open*, 1(1), p.9.

Buttfield, A., Ferrez, P.W. & Millán, J.D.R., 2006. Towards a Robust BCI: Error Potentials and Online Learning. *IEEE Trans Neural Rehab Sys Eng*, 14(2), pp.164–168.

Carter, C.S. et al., 1998. Anterior Cingulate Cortex, Error Detection, and the Online Monitoring of Performance. *Science*, 280(5364), pp.747–749.

Casanova, E. et al., 2003. Locked-in Syndrome: Improvement in the Prognosis After an Early Intensive Multidisciplinary Rehabilitation. *Arch Phys Med Rehabil*, 84(6), pp.862–867.

Cavanagh, J.F. et al., 2010. Frontal theta links prediction errors to behavioral adaptation in reinforcement learning. *NeuroImage*, 49(4), pp.3198–3209.

Chaudhary, U., Birbaumer, N. & Ramos-Murguialday, A., 2016. Brain-computer interfaces in the completely locked-in state and chronic stroke. In *Progress in Brain Research*. Elsevier B.V., pp. 131–161.

Chavarriaga, R., Ferrez, P. & Millán, J., 2007. To err is human: Learning from error potentials in brain-computer interfaces. In *Proceedings of the 1st International Conference on Cognitive Neurodynamics (ICCN 2007)*. Shanghai, China, pp. 1–4.

Chavarriaga, R. & Millán, J.D.R., 2010. Learning From EEG Error-Related Potentials in Noninvasive Brain-Computer Interfaces. *IEEE Trans Neural Rehab Sys Eng*, 18(4), pp.381–388.

Chavarriaga, R., Sobolewski, A. & Millán, J.D.R., 2014. Errare machinale est: the use of error-related potentials in brain-machine interfaces. *Frontiers in Neuroscience*, 8(July), p.13.

Cohen, M.X., 2011. Error-related medial frontal theta activity predicts cingulate-related structural connectivity. *NeuroImage*, 55(3), pp.1373–1383.

Congedo, M., Barachant, A. & Andreev, A., 2013. *A New Generation of Brain-Computer Interface Based on Riemannian Geometry*, arXiv preprint arXiv:1310.8115.

Dangi, S. et al., 2013. Design and analysis of closed-loop decoder adaptation algorithms for brain-machine interfaces. *Neural Computation*, 25(7), pp.1693–1731.

Delorme, A. & Makeig, S., 2004. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J Neurosc Methods*, 134(1), pp.9–21.

Doble, J.E. et al., 2003. Impairment, activity, and survival in persons with locked-in syndrome for over a decade. *J Head Trauma Rehabil*, 18(5), pp.435–444.

Emeric, E.E. et al., 2008. Performance Monitoring Local Field Potentials in the Medial Frontal Cortex of Primates: Anterior Cingulate Cortex. *J Neurophysiol*, 99, pp.759–772.

Emeric, E.E. et al., 2010. Performance Monitoring Local Field Potentials in the Medial Frontal Cortex of Primates: Supplementary Eye Field. *J Neurophysiol*, 104(3), pp.1523–1537.

Falkenstein, M. et al., 1991. Effects of crossmodal divided attention on late ERP components. II. Error processing in choice reaction tasks. *Electroencep. Clin. Neurophysiol.*, 78, pp.447–455.

Falkenstein, M. et al., 2000. ERP components on reaction errors and their functional significance: a tutorial. *Biological Psychology*, 51, pp.87–107.

Farwell, L.A. & Donchin, E., 1988. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencep. Clin. Neurophysiol*, 70(6), pp.510–523.

Ferrez, P.W. & Millan, J. del R., 2008. Error-Related EEG Potentials Generated During Simulated Brain-Computer Interaction. In *IEEE Trans Biomed Eng*. pp. 923–929.

Ferrez, P.W. & Millán, J. del R., 2005. You Are Wrong! - Automatic Detection of Interaction Errors from Brain Waves. In *Conf Proc IJCAI*. Edinburg, U.K.

Friehs, G.M. et al., 2004. Brain-Machine and Brain-Computer Interfaces. *Stroke*, 35(11 Suppl 1), pp.2702–2705.

Gehring, W.J. et al., 1993. A neural system for error detection and compensation. *Psychol. Science*, 4(6), pp.385–390.

Geng, S. et al., 2013. Extraction of Error Related Local Field Potentials from the Striatum during Environmental Perturbations of a Robotic Arm. In *Conf Proc IEEE/EMBS on Neural Engineering*. San Diego, California, USA, pp. 993–996.

Godlove, D.C. et al., 2011. Event-Related Potentials Elicited by Errors during the Stop-Signal Task. I. Macaque Monkeys. *J Neurosc.*, 31(44), pp.15640–9.

Goodrich, M. a. & Schultz, A.C., 2007. Human-Robot Interaction: A Survey. *Foundations and Trends in Human-Computer Interaction*, 1(3), pp.203–275.

Guger, C. et al., 2012. How many people could use an SSVEP BCI? *Frontiers in Neuroscience*, 6, p.6.

Guger Technologies, [Online], 2016. Guger technologies. Available at: http://www.gtec.at/.

Gürel, T. & Mehring, C., 2012. Unsupervised adaptation of brain-machine interface decoders. *Frontiers in Neuroscience*, 6, p.18.

Guyton, A.C. & Hall, J.E., 1990. *Textbook of Medical Physiology* 8th Ed. A. C. Guyton & J. E. Hall, eds., Harcourt College Publishers.

Haig, A., Katz, R. & Sahgal, V., 1987. Mortality and complications in the locked-in syndrome. *Arch Phys Med Rehabil*, 68, pp.24–27.

Hajcak, G., McDonald, N. & Simons, R.F., 2003. To err is autonomic: Error-related brain potentials, ANS activity, and post-error compensatory behavior. *Psychophysiology*, 40(6), pp.895–903.

Hewig, J. et al., 2011. Dissociation of Pe and ERN/Ne in the conscious recognition of an error. *Psychophysiology*, 48(10), pp.1390–1396.

Higger, M. et al., 2014. A Bayesian Framework for Intent Detection and Stimulation Selection in SSVEP BCIs. *IEEE Signal Processing Letters*, 22(6), pp.743–747.

Hoffmann, U., Vesin, J.-M. & Ebrahimi, T., 2006. Spatial filters for the classification of event-related potentials. In *14th ESANN*. Bruges, Belgium, pp. 47–52.

Holroyd, C.B. & Coles, M.G.H., 2002. The Neural Basis of Human Error Processing: Reinforcement Learning, Dopamine, and the Error-Related Negativity. *Psychological Review*, 109(4), pp.679–709.

Iturrate, I. et al., 2009. A Noninvasive Brain-Actuated Wheelchair Based on a P300

Neurophysiological Protocol and Automated Navigation. *IEEE Trans Robotics*, 25(3), pp.614–627.

Iturrate, I. et al., 2014. Latency correction of event-related potentials between different experimental protocols. *J Neural Engineering*, 11, p.12.

Iturrate, I., Montesano, L. & Mínguez, J., 2010. Robot Reinforcement Learning using EEG-based reward signals. In *Conf Proc IEEE Robots Automat*. Anchorage, Alaska, USA, pp. 4822–4829.

Jia, N. et al., 2017. Decoding of intended saccade direction in an oculomotor brain-computer interface. *J Neural Engineering*, Accepted, p.11.

Katzner, S. et al., 2009. Local Origin of Field Potentials in Visual Cortex. *Neuron*, 61(1), pp.35–41.

Klein, T.A. et al., 2007. Neural correlates of error awareness. *NeuroImage*, 34(4), pp.1774–1781.

Kosuge, K. & Hirata, Y., 2004. Human-Robot Interaction. In *Conf Proc IEEE/ROBIO 2004*. Shenyang, China, pp. 8–11.

Krusienski, D.J. & Shih, J.J., 2011. Control of a visual keyboard using an electrocorticographic brain-computer interface. *Neurorehabil and Neural Repair*, 25(4), pp.323–331.

Kübler, A. et al., 2013. Facing the challenge: Bringing brain-computer interfaces to end-users. *Artificial Intelligence in Medicine*, 59(2), pp.55–60.

Kübler, A. et al., 2005. Severity of Depressive Symptoms and Quality of Life in Patients with Amyotrophic Lateral Sclerosis. *Neurorehabil and Neural Repair*, 19(3), pp.182–193.

Kübler, A. & Birbaumer, N., 2008. Brain-computer interfaces and communication in paralysis: Extinction of goal directed thinking in completely paralysed patients? *Clinical Neurophysiology*, 119(11), pp.2658–2666.

Kushki, A. et al., 2012. Classification of Activity Engagement in Individuals with Severe Physical Disabilities Using Signals of the Peripheral Nervous System. *PloS One*, 7(2), p.8.

LaFleur, K. et al., 2013. Quadcopter control in three-dimensional space using a noninvasive motor imagery based brain-computer interface. *J Neural Engineering*, 10(4), p.15.

Leuthardt, E.C. et al., 2006. The Emerging World of Motor Neuroprosthetics: A

Neurosurgical Perspective. *Neurosurgery*, 58(7), pp.1–14.

Llera, A. et al., 2011. On the use of interaction error potentials for adaptive brain computer interfaces. *Neural Networks*, 24(10), pp.1120–1127.

Lopez-Larraz, E. et al., 2010. Real-Time Recognition of Feedback Error-Related Potentials during a Time-Estimation Task. In *Conf Proc IEEE/EMBS*. pp. 2670–2673.

Lotte, F. & Guan, C., 2011. Regularizing Common Spatial Patterns to Improve BCI Designs: Unified Theory and New Algorithms. *IEEE Trans Biomed Eng*, 58(2), pp.355–362.

Lutz, A. et al., 2008. Attention regulation and monitoring in meditation. *Trends in Cogn Sci*, 12(4), pp.163–169.

Luu, S. & Chau, T., 2009. Decoding subjective preference from single-trial near-infrared spectroscopy signals. *J Neural Engineering*, 6(1), p.8.

Manna, A. et al., 2010. Neural correlates of focused attention and cognitive monitoring in meditation. *Brain Research Bulletin*, 82, pp.46–56.

Masimore, B., Kakalios, J. & Redish, A.D., 2004. Measuring fundamental frequencies in local field potentials. *J Neurosc Methods*, 138, pp.97–105.

McFarland, D.J. et al., 1997. Spatial filter selection for EEG-based communication. *Electroencep. Clin. Neurophysiol*, 103(3), pp.386–94.

McFarland, D.J., Sarnacki, W.A. & Wolpaw, J.R., 2011. Should the parameters of a BCI translation algorithm be continually adapted? *J Neurosc Methods*, 199, pp.103–107.

Mellinger, J. et al., 2007. An MEG-based brain-computer interface (BCI). *NeuroImage*, 36(3), pp.581–593.

Milekovic, T. et al., 2012. Error-related electrocorticographic activity in humans during continuous movements. *J Neural Engineering*, 9(2), p.21.

Millán, J. de. R. et al., 2010. Combining brain-computer interfaces and assistive technologies: state-of-the-art and challenges. *Frontiers in Neuroscience*, 4(September), p.15.

Miltner, W.H.R., Braun, C.H. & Coles, M.G.H., 1997. Event-Related Brain Potentials Following Incorrect Feedback in a Time-Estimation Task: Evidence for a “Generic” Neural System for Error Detection. *J Cogn Neurosc*, 9(6), pp.788–798.

Naci, L. et al., 2012. Brain-computer interfaces for communication with nonresponsive patients. *Annals of Neurology*, 72(3), pp.312–323.

New, P.W. & Thomas, S.J., 2005. Cognitive impairments in the locked-in syndrome: A case report. *Arch Phys Med Rehabil*, 86(2), pp.338–343.

Nieuwenhuis, S. et al., 2001. Error-related brain potentials are differentially related to awareness of response errors: Evidence from an antisaccade task. *Psychophysiology*, 38(5), pp.752–760.

O'Connell, R.G. et al., 2007. The role of cingulate cortex in the detection of errors with and without awareness: a high-density electrical mapping study. *European Journal of Neuroscience*, 25(8), pp.2571–2579.

Parra, L.C. et al., 2005. Recipes for the linear analysis of EEG. *NeuroImage*, 28(2), pp.326–341.

Parra, L.C. et al., 2008. Spatiotemporal Linear Decoding of Brain State: Application to performance augmentation in high-throughput tasks. *IEEE Signal Processing Magazine*, 25(1), pp.107–115.

Patterson, J.R. & Grabois, M., 1986. Locked-in syndrome: a review of 139 cases. *Stroke*, 17(4), pp.758–764.

Perelmouter, J. & Birbaumer, N., 2000. A Binary Spelling Interface with Random Errors. *IEEE Trans Rehab Eng*, 8(2), pp.227–232.

Perrin, X. et al., 2010. Brain-coupled interaction for semi-autonomous navigation of an assistive robot. *Robotics and Autonomous Systems*, 58(12), pp.1246–1255.

Pesaran, B., 2009. Uncovering the Mysterious Origins of Local Field Potentials. *Neuron*, 61(1), pp.1–2.

Prins, N.W. et al., 2013. Feature Extraction and Unsupervised Classification of Neural Population Reward Signals for Reinforcement Based BMI. In *35th Conf Proc IEEE/EMBS*. Osaka, Japan, pp. 5250–5253.

Ramli, R. et al., 2015. Using finite state machine and a hybrid of EEG signal and EOG artifacts for an asynchronous wheelchair navigation. *Expert Systems with Applications*, 42(5), pp.2451–2463.

Regan, D., 1989. *Human Brain Electrophysiology: Evoked Potentials and Evoked Magnetic Fields in Science and Medicine* 1st Ed. D. Regan, ed., New York: Elsevier Science Ltd.

Rivet, B. et al., 2011. Theoretical Analysis of XDAWN Algorithm: Application to an Efficient Sensor Selection in a P300 BCI. In *19th Conf Proc EUSIPCO*. Barcelona, Spain, pp. 1382–1386.

Rivet, B. et al., 2009. xDAWN Algorithm to Enhance Evoked Potentials: Application to Brain-Computer Interface. *IEEE Trans Biomed Eng*, 56(8), pp.2035–2043.

Rousseaux, M. et al., 2009. Evidence of persisting cognitive impairment in a case series of patients with locked-in syndrome. *J Neurol Neurosurg Psychiatry*, 80, pp.166–170.

Sanchez, J.C. et al., 2009. Exploiting co-adaptation for the design of symbiotic neuroprosthetic assistants. *Neural Networks*, 22(3), pp.305–315.

Santhanam, G. et al., 2006. A high-performance brain-computer interface. *Nature*, 442(7099), pp.195–198.

Schalk, G. et al., 2000. EEG-based communication: presence of an error potential. *Clinical Neurophysiology*, 111(12), pp.2138–2144.

Scheffers, M.K. & Coles, M.G.H., 2000. Performance Monitoring in a Confusing World: Error-Related Brain Activity, Judgments of Response Accuracy, and Types of Errors. *J Exp Psych*, 26(1), pp.141–151.

van Schie, H.T. et al., 2004. Modulation of activity in medial frontal and motor cortices during error observation. *Nature Neuroscience*, 7(5), pp.549–554.

Schnakers, C. et al., 2008. Cognitive function in the locked-in syndrome. *J Neurology*, 255(3), pp.323–330.

Sellers, E.W. & Donchin, E., 2006. A P300-based brain-computer interface: Initial tests by ALS patients. *Clinical Neurophysiology*, 117(3), pp.538–548.

Shenoy, P. et al., 2006. Towards adaptive classification for BCI. *J Neural Engineering*, 3(1), pp.R13–R23.

Sitaram, R. et al., 2007. Temporal classification of multichannel near-infrared spectroscopy signals of motor imagery for developing a brain-computer interface. *NeuroImage*, 34(4), pp.1416–1427.

Smith, E. & Delargy, M., 2005. Clinical review. Locked-in syndrome. *BMJ Open*, 330(February), pp.406–409.

Spüler, M., Bensch, M., et al., 2012. Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a P300-BCI. *Clinical Neurophysiology*, 123(7), pp.1328–1337.

Spüler, M., Rosenstiel, W. & Bogdan, M., 2012. Online Adaptation of a c-VEP Brain-computer Interface(BCI) Based on Error-Related Potentials and Unsupervised Learning. *PloS One*, 7(12), p.11.

Stoll, J. et al., 2013. Pupil responses allow communication in locked-in syndrome patients. *Current Biology*, 23(15), pp.R647–R648.

Tan, P.-N., Steinbach, M. & Kumar, V., 2005. Cluster Analysis: Basic Concepts and Algorithms. In P.-N. Tan, M. Steinbach, & V. Kumar, eds. *Introduction to Data Mining*. Addison-Wesley, pp. 487–568.

Tonin, L. et al., 2011. Brain-Controlled Telepresence Robot by Motor-Disabled People. In *33rd Conf Proc IEEE/EMBS*. Boston, MA, USA, pp. 4227–4230.

Ullsperger, M., Danielmeier, C. & Jocham, G., 2014. Neurophysiology of Performance Monitoring and Adaptive Behavior. *Physiological Reviews*, 94(1), pp.35–79.

Vidaurre, C. et al., 2011. Co-adaptive calibration to improve BCI efficiency. *J Neural Engineering*, 8(2), p.8.

Vocat, R., Pourtois, G. & Vuilleumier, P., 2008. Unavoidable errors: A spatio-temporal analysis of time-course and neural sources of evoked potentials associated with error processing in a speeded task. *Neuropsychologia*, 46(10), pp.2545–2555.

Voytek, B. et al., 2013. A method for event-related phase/amplitude coupling. *NeuroImage*, 64(1), pp.416–424.

Wolpaw, J.R. et al., 2002. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6), pp.767–791.

Wolpaw, J.R. et al., 1998. EEG-based Communication: Improved Accuracy by Response Verification. *IEEE Trans Rehab Eng*, 6(3), pp.326–333.

Yordanova, J. et al., 2004. Parallel systems of error processing in the brain. *NeuroImage*, 22(2), pp.590–602.

Zhang, H. et al., 2015. EEG-based decoding of error-related brain activity in a real-world driving task. *J Neural Engineering*, 12(6), p.10.

CURRICULUM VITAE

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]