



HAL
open science

Interfaces Cerveau-Machines basées sur l'imagination de mouvements brefs : vers des boutons contrôlés par la pensée

Joan Fruitet

► To cite this version:

Joan Fruitet. Interfaces Cerveau-Machines basées sur l'imagination de mouvements brefs : vers des boutons contrôlés par la pensée. Traitement du signal et de l'image [eess.SP]. Université Nice Sophia Antipolis, 2012. Français. NNT: . tel-00838704

HAL Id: tel-00838704

<https://theses.hal.science/tel-00838704v1>

Submitted on 26 Jun 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITE DE NICE - SOPHIA ANTIPOLIS
ECOLE DOCTORALE STIC
SCIENCES ET TECHNOLOGIES DE L'INFORMATION
ET DE LA COMMUNICATION

THESE DE DOCTORAT

Spécialité : Automatique, Traitement du signal et des images

Présentée par

Joan FRUITET

Interfaces Cerveau-Machines basées sur l'imagination de mouvements brefs : vers des boutons contrôlés par la pensée

préparée à l'INRIA Sophia Antipolis, équipe-projet ATHÉNA

sous la direction de
Maureen CLERC

soutenue le 4 juillet 2012, devant les membres du jury :

<i>Rapporteurs :</i>	Christian JUTTEN	- GIPSA-lab
	Anatole LÉCUYER	- INRIA Equipe VR4I
<i>Examineurs :</i>	Jérémie MATTOU	- INSERM U1028
	David GUIRAUD	- INRIA EPI DEMAR
	Théodore PAPADOPOULO	- INRIA EPI Athéna
<i>Directrice de thèse :</i>	Maureen CLERC	- INRIA EPI Athéna

RÉSUMÉ

Les Interfaces Cerveau Machine (ICM) sont des dispositifs d'un type nouveau permettant la communication directe entre le cerveau d'un utilisateur et une machine. De tels dispositifs peuvent être réalisés grâce à la mesure non-invasive d'informations provenant du cortex par électroencéphalographie (EEG). Un des enjeux principaux du domaine est de réussir à extraire en temps réel, à partir d'une source d'information très limitée et bruitée, un signal de commande robuste et suffisamment complexe pour permettre le contrôle d'un programme ou d'un effecteur.

Dans cette thèse nous avons contribué à l'amélioration des ICM sur trois points. Tout d'abord, nous avons exploré la possibilité d'augmenter la résolution spatiale de l'EEG en utilisant des méthodes permettant de construire l'activité corticale en temps réel. D'autre part, inspirés par une ICM utilisant l'imagination motrice des pieds pour envoyer une commande unique, nous nous sommes intéressés aux signaux produits par l'imagination de mouvements brefs. Cela nous a permis de développer un nouveau type d'ICM appelé bouton commandé par la pensée. Une telle ICM permet à l'utilisateur d'envoyer, de façon asynchrone, plusieurs commandes en imaginant différents mouvements. Finalement, nous avons développé une méthode, basée sur la théorie des bandits stochastiques, pour sélectionner automatiquement et efficacement le mouvement imaginé le plus discernable de l'état de repos et permettant ainsi le meilleur contrôle de l'ICM. En parallèle, nous avons développé une boîte à outils Matlab qui automatise l'ajustement des paramètres et la comparaison des différentes méthodes utilisées pour réaliser une ICM.

ABSTRACT

Brain Computer Interfaces are a new type of device, allowing direct communication between a user's brain and a machine. Such devices have been proposed based on non-invasive brain activity measurements using electroencephalography (EEG). One of the challenges in this domain is to extract in real time a sufficiently complex and robust signal from a very limited and noisy source of information, in order to control a program or an assistive device.

This thesis explores three aspects of Brain Computer Interfaces. The first goal is to enhance the spatial resolution of EEG using source localization methods. These methods can reconstruct detailed cortical activity based on non-invasive scalp measurements. The second aspect is inspired by a BCI that uses foot motor imagination to send a unique command. This started with the study of short movement imaginations and led to a new type of BCI called brain-controlled buttons. The BCI we created enables the user to send commands, in a self paced manner, by imagining different movements. Finally, we develop a method, based on stochastic bandits theory, to automatically and efficiently select between different types of imagined movements those that are the most detectable from a resting state and thus the most appropriate to control a BCI. In parallel, we developed a Matlab toolbox that automates, via off-line analysis, the comparison of different methods and the selection of all the parameters used in a BCI.

Table des matières

I	Tout ce que vous avez toujours voulu savoir sur les ICM	11
1	Principes de base du fonctionnement d'une ICM	13
1.1	Qu'est ce qu'une ICM ?	13
1.2	Fonctionnement d'une ICM	13
1.3	Terminologie sur les différents types d'ICM	19
2	Quelques notions de neurophysiologie	19
2.1	Fonctionnement d'un neurone	19
2.2	Que peut-on espérer enregistrer de l'activité neuronale ?	21
3	Les méthodes d'acquisition	22
3.1	Méthodes d'acquisition invasives	22
3.2	Méthodes d'acquisition non invasives	25
3.3	Pourquoi avons-nous choisi l'EEG ?	29
4	Les ICM utilisant l'EEG	30
4.1	ICM basées sur les potentiels évoqués	30
4.2	ICM utilisant des signaux spontanés	39
II	État de l'art de quelques problématiques liées aux ICM	47
1	Introduction	49
1.1	Nouvelles ICM et interfaces de contrôle	49
1.2	Extraction du signal de contrôle	49
2	Différentes façons d'utiliser l'imagination motrice	50
2.1	Contrôle multidimensionnel modulé en amplitude	51
2.2	Utilisation synchrone de l'imagination motrice	52
2.3	Détection asynchrone de tâches	52
2.4	Sélection de tâches adaptées au contrôle d'une ICM	56
3	Augmentation de la résolution spatiale de l'EEG	58
3.1	Filtrage spatial laplacien	58
3.2	Le <i>Common Spatial Filter</i> ou CSP	59
3.3	Reconstruction de sources	61
III	Imagination motrice et rythmes sensori-moteurs	67
1	Classification de mouvements continus	69
1.1	Enregistrements EEG durant des mouvements continus	69
1.2	Traitement des données	75
1.3	Résultats	78
1.4	Discussion	80
2	ICM utilisant des mouvements brefs	82
2.1	Le rebond bêta	82

2.2	ICM de navigation spatiale basée sur le rebond bêta	84
2.3	Résultats	89
2.4	Perspectives	90
3	Étude détaillée des mouvements brefs	92
3.1	Objectifs	92
3.2	Enregistrements EEG de mouvements brefs	93
3.3	Analyse temps-fréquence des données	97
3.4	Traitement hors-ligne des données	106
3.5	Détection et discrimination en-ligne de deux tâches	108
3.6	Conclusion et Perspectives	109
4	Conclusion	113
IV	Sélection automatique de tâches cognitives	115
1	La sélection des tâches contrôlant les ICM	117
1.1	Pourquoi faire de la sélection de tâches?	117
1.2	Un problème exigeant des compromis	117
2	Initiation à la théorie des bandits	118
2.1	La problématique du bandit manchot	118
2.2	Le compromis exploration/exploitation	119
3	Sélection automatique d'une tâche optimale	120
3.1	Modélisation de notre problème	120
3.2	L'algorithme de sélection de tâches	120
3.3	Évaluation de performances sur un nombre faible de réalisations	124
4	Analyse hors-ligne de l'algorithme <i>UCB-classif</i>	127
4.1	Objectifs et méthode pour l'analyse hors-ligne	127
4.2	Résultats hors-ligne	128
5	Sélection d'une tâche optimale en-ligne	131
5.1	Scénario OpenViBE pour la sélection de tâches en-ligne	131
5.2	Résultats	133
6	Conclusion et perspectives	135
V	Reconstruction de sources pour les ICM	137
1	Pourquoi reconstruire l'activité des sources corticales?	139
1.1	Augmenter la résolution spatiale de l'EEG	139
1.2	Obtenir des caractéristiques ayant un sens physiologique	140
2	Des sources aux capteurs : le problème direct	140
2.1	Modélisation de l'activité corticale	141
2.2	Modélisation anatomique	142
2.3	Résolution du problème direct	143
3	Des capteurs aux sources : le problème inverse	144
3.1	Résolution par Minimum-Norm	144
3.2	Résolution par filtrage : le "beamformer"	146

4	Utilisation de la reconstruction de l'activité corticale pour les ICM	151
4.1	Modification de la chaîne de traitement d'une ICM	151
4.2	Sélection de features	152
4.3	Coût de l'utilisation de la reconstruction de sources	153
5	Résultats	154
5.1	Comparaison des méthodes sur des données d'imagination motrice	154
5.2	Discussion	156
6	Conclusion	159
VI	Toolbox Matlab pour les ICM	161
1	<i>MoffBCI</i>, une nouvelle toolbox pour les ICM	163
1.1	Pourquoi une nouvelle toolbox pour les ICM?	163
1.2	Cahier des charges de <i>MoffBCI</i>	165
2	Fonctionnement de <i>MoffBCI</i>	167
2.1	Les fonctions de traitement	167
2.2	La définition d'une chaîne de traitement	167
2.3	Quelques détails sur les propriétés du noyau	168
3	Méthodes d'évaluation en apprentissage	171
3.1	Quelques notions sur l'apprentissage supervisé	171
3.2	Les méthodes d'évaluation de <i>MoffBCI</i>	173
VII	Conclusion	179
	Annexes	183
A	Exemple d'utilisation de <i>MoffBCI</i>	185
1	Génération d'un jeu de données artificiel	185
2	Écriture d'une chaîne de traitement	185
3	Évaluation de la chaîne de traitement	186
3.1	Évaluation standard	186
3.2	Validation croisée	187
3.3	Évaluation tripartite ou <i>3-way data split</i>	187
3.4	Double validation croisée	187
4	Résultats	188
B	Écriture d'une fonction de traitement de <i>MoffBCI</i>	193
1	Exemple : la fonction "temporal filter"	193
1.1	Définition	193
1.2	Les entrées et sorties	193
1.3	Les paramètres	193

Bibliographie

195

Introduction

UNE Interface Cerveau Machine (ICM) est une interface permettant d'envoyer des informations directement à partir de l'activité cérébrale vers le monde extérieur, sans passer par la voie naturelle des nerfs et des muscles. Depuis que le Dr Vidal a prouvé qu'un tel dispositif était réalisable [Vidal, 1973], une communauté scientifique grandissante s'attèle à améliorer les capacités des ICM. Il reste cependant de nombreux verrous qui devront être débloqués pour rendre les ICM réellement utilisables.

Le travail que nous avons effectué porte sur les ICM non invasives utilisant l'électroencéphalographie (EEG) comme méthode d'enregistrement et plus particulièrement sur les ICM basées sur l'imagination d'actions motrices. Plutôt que de présenter le travail réalisé durant cette thèse dans l'ordre chronologique, nous avons choisi d'organiser ce mémoire sous une forme plus didactique.

Chapitre I : Tout ce que vous avez toujours voulu savoir sur les ICM

Le premier chapitre de ce manuscrit est une présentation générale du domaine des Interfaces Cerveau Machine. Nous y décrivons le fonctionnement d'une ICM dans son ensemble et nous définissons une terminologie précise. Après un aperçu de l'origine physiologique des signaux corticaux, un état de l'art des méthodes d'enregistrement puis des protocoles utilisés pour les ICM est effectué.

Chapitre II : État de l'art de quelques problématiques liées aux ICM

Un état de l'art plus approfondi, portant sur les sujets que nous avons abordés durant ce travail, est effectué dans ce chapitre. Après avoir rapidement présenté

le cadre de la recherche portant sur les ICM, nous détaillons les différentes utilisations de l'imagination motrice dans les ICM. En particulier nous redéfinissons le concept de *boutons commandés par la pensée*, dont l'étude sera développée dans le chapitre III et qui est l'une des contributions centrales de ce travail de thèse. Nous soulignons également l'importance de la sélection de tâches d'imagination motrice appropriées pour le contrôle d'une ICM. Ceci introduit le travail présenté dans le chapitre IV sur la sélection automatique de tâches. Enfin, un état de l'art portant sur les méthodes permettant d'augmenter la résolution de l'EEG est effectué avec un accent sur les différentes méthodes permettant de reconstruire l'activité corticale. Les détails du fonctionnement de certaines de ces méthodes seront données dans le chapitre V.

Chapitre III : Imagination motrice et rythmes sensori-moteurs

Ce chapitre regroupe les différentes expériences que nous avons développées pour l'étude des ICM basées sur l'imagination motrice. La première partie concerne le travail effectué au début de cette thèse sur l'imagination de mouvements moteurs continus soutenus. Ce travail, s'il n'est pas novateur, nous a permis de développer une chaîne de traitement complète et de prendre conscience des difficultés tant théoriques que pratiques. Nous avons utilisé et contribué au logiciel open-source OpenViBE, spécialement développée pour réaliser des ICM [Renard et al., 2010]. Les données recueillies ont servi de base à l'étude de la reconstruction de l'activité corticale pour les ICM qui fait l'objet du chapitre V.

La seconde partie de ce chapitre est consacrée à une application de navigation spatiale basée sur un unique bouton contrôlé par la pensée. Ce travail, réalisé avec l'ingénieur Nicolas Servant, a permis de réutiliser un bouton contrôlé par l'imagination motrice des pieds (plus précisément par le rebond bêta) disponible dans OpenViBE pour se déplacer dans un environnement virtuel. Il s'agit donc d'un travail effectué sur la conversion d'un signal de contrôle extrêmement réduit (binaire) vers une commande plus complexe (navigation spatiale).

Malgré la simplicité du transducteur utilisé pour détecter les rebonds bêta, nous avons obtenu des résultats très encourageants. Ceci nous a poussé à améliorer le transducteur pour permettre non seulement une détection plus fiable mais aussi augmenter le nombre de contrôles possibles. Pour cela nous avons étudié de façon approfondie l'imagination des mouvements brefs. Cette étude, qui fait l'objet de la dernière partie de ce chapitre, est l'une des contributions majeures de cette thèse et a permis de prouver qu'il est possible de contrôler plusieurs boutons par la pensée grâce à l'imagination de mouvements brefs.

Chapitre IV : Sélection automatique de tâches cognitives

Ce chapitre est consacré à la sélection de tâches pour le contrôle des ICM. Après avoir rappelé l'importance d'utiliser des tâches adaptées aux sujets, nous expliquons le fonctionnement d'une méthode automatique pour la sélection de tâches pour les ICM, développée en collaboration avec Alexandra Carpentier. Afin d'explorer de façon optimale les différentes tâches proposées, nous avons utilisé la théorie des bandits stochastiques. Une preuve théorique des performances est donnée ainsi que deux études expérimentales. La première utilise les données d'imagination motrice brève décrites dans le chapitre III, pour une analyse hors ligne des performances. La seconde consiste en une vérification en ligne des capacités de l'algorithme à sélectionner une tâche optimale.

Chapitre V : Reconstruction de sources pour les ICM

Ce chapitre décrit le fonctionnement des méthodes de reconstruction des sources corticales pour les ICM et plus particulièrement de deux méthodes :

- la résolution par Minimum-Norm,
- la résolution par filtrage de type “beamformer”.

Ce travail a été réalisé en début de thèse en utilisant les données d'imagination motrice continue décrites dans la première partie du chapitre III. Il montre une nouvelle fois qu'il est possible d'utiliser en temps réel des méthodes de reconstruction de sources, mais à cause du nombre limité d'expériences et de sujets, il ne permet pas de conclure sur une réelle utilité de telles méthodes pour améliorer les performances des ICM.

Chapitre VI : Toolbox Matlab pour les ICM

L'ensemble des résultats présentés dans les chapitres III, IV et V ont été obtenus grâce à une boîte à outils Matlab, appelé *MoffBCI*, que nous avons développée spécialement pour l'analyse et l'optimisation hors ligne des ICM. Cette boîte à outils offre un bon complément à OpenViBE et nous pensons donc qu'elle pourrait intéresser de nombreuses personnes. Ce chapitre décrit le fonctionnement de *MoffBCI* ainsi que les diverses fonctions qui y sont implémentées. Pour plus de détails, l'annexe A donne un exemple simple d'utilisation et l'annexe B est consacré à l'implémentation d'une fonction de la boîte à outils.

Tout ce que vous avez toujours voulu savoir sur les ICM (ou presque)

“Au commencement était le Verbe, et le Verbe était [Signal]”¹

L E but de ce chapitre est de permettre à un lecteur non spécialiste des interfaces cerveau-machine de comprendre ce qui sera exposé dans la suite de ce manuscrit.

Les personnes amenées à s’intéresser aux ICM étant d’origines très diverses², nous avons jugé qu’il était préférable de ne pas rentrer dans les détails techniques, mais plutôt d’essayer de donner une vision d’ensemble du domaine. Le chapitre suivant abordera certaines problématiques avec plus de précisions.

¹évangile [des ICM] selon St J[o]an.

²Durant ma thèse j’ai rencontré des biologistes spécialisés dans les neurosciences, des mathématiciens experts du traitement du signal, des informaticiens férus d’intelligence artificielle et même des gardiens de nuit quand on reste à rédiger sa thèse un peu trop tard...

Sommaire

1	Principes de base du fonctionnement d'une ICM	13
1.1	Qu'est ce qu'une ICM?	13
1.2	Fonctionnement d'une ICM	13
1.2.1	<i>Acquisition du signal</i>	14
1.2.2	<i>Extraction du signal de contrôle</i>	15
1.2.3	<i>Conversion du signal de contrôle en commandes</i>	16
1.2.4	<i>Le Feedback</i>	18
1.3	Terminologie sur les différents types d'ICM	19
2	Quelques notions de neurophysiologie	19
2.1	Fonctionnement d'un neurone	19
2.2	Que peut-on espérer enregistrer de l'activité neuronale?	21
3	Les méthodes d'acquisition	22
3.1	Méthodes d'acquisition invasives	22
3.1.1	<i>ECoG</i>	23
3.1.2	<i>Électrodes implantées</i>	23
3.2	Méthodes d'acquisition non invasives	25
3.2.1	<i>EEG</i>	25
3.2.2	<i>MEG</i>	27
3.2.3	<i>IRMf</i>	29
3.2.4	<i>NIRS</i>	29
3.3	Pourquoi avons-nous choisi l'EEG?	29
4	Les ICM utilisant l'EEG	30
4.1	ICM basées sur les potentiels évoqués	30
4.1.1	<i>ICM utilisant le P300</i>	30
4.1.2	<i>Steady State Visual Evoked Potential (SSVEP)</i>	35
4.1.3	<i>Potentiels d'erreur</i>	37
4.2	ICM utilisant des signaux spontanés	39
4.2.1	<i>Ondes corticales lentes</i>	39
4.2.2	<i>Imagination motrice et rythmes sensori-moteurs</i>	40
4.2.3	<i>Reconnaissance de tâches cognitives</i>	45

1 PRINCIPES DE BASE DU FONCTIONNEMENT D'UNE ICM

1.1 Qu'est ce qu'une ICM ?

Une interface cerveau machine (ICM, ou BCI pour *brain computer interface*) est un dispositif permettant une communication directe entre le cerveau d'un utilisateur et une machine, ou un ordinateur, sans passer par les voies naturelles constituées de la moelle épinière, des nerfs et des muscles³.

Cela fait presque 30 ans que le Dr Vidal a prouvé qu'un tel concept était réalisable [Vidal, 1973], et pourtant les performances actuelles des ICM sont encore extrêmement faibles. A titre d'exemple le P300Speller, qui est une ICM destinée à simuler un clavier, permet de taper environ 8 lettres par minute [Donchin *et al.*, 2000].

L'objectif à court terme des ICM n'est donc pas de remplacer le clavier de tout un chacun, mais de permettre à des personnes souffrant de lourdes paralysies de retrouver un moyen de communication avec le monde extérieur.

Pendant les ICM peuvent être vues à moyen terme comme un complément des interfaces traditionnelles (clavier, souris...). Plusieurs firmes (www.emotiv.com, www.neurosky.com) développent même déjà des ICM à des fins vidéo-ludiques.

1.2 Fonctionnement d'une ICM

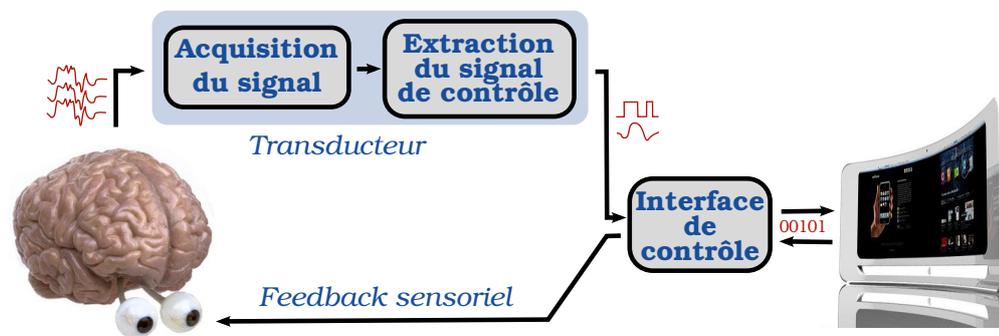


Fig. I.1 – Schéma de fonctionnement d'une interface cerveau-machine

Comme présenté sur le schéma I.1, le fonctionnement d'une ICM peut être découpé en quatre étapes qui seront détaillées plus bas :

³Nous nous restreindrons dans ce manuscrit aux ICM, parfois appelé "de sortie" [Besserve, 2007], permettant une communication du cerveau vers le monde extérieur. Les dispositifs permettant une transition des informations du monde extérieur vers le cerveau (tel que des prothèses auditives ou visuelles) ne sont pas, à mon sens, des ICM puisqu'elles ne permettent pas une communication vers le monde extérieur.

1. **Acquisition du signal.** On mesure l'activité cérébrale et on la convertit en signal numérique.
2. **Extraction du signal de contrôle à partir du signal biologique.** Il s'agit d'extraire du signal biologique des commandes brutes, appelées signal de contrôle [Mason *et al.*, 2007]. C'est la partie centrale d'une ICM.
3. **Interface de contrôle** Il s'agit de traduire (et éventuellement d'optimiser) le signal de contrôle en commandes et de contrôler une machine, ou un programme. Une commande intelligente est primordiale pour qu'une ICM soit efficace voire simplement utilisable.
4. **Production du feedback.** La dernière étape de fonctionnement d'une ICM est la production d'un retour sensitif (ou feedback). Ce retour, généralement visuel, est extrêmement important puisqu'il permet à l'utilisateur d'apprendre à contrôler l'interface.

1.2.1 Acquisition du signal

Les différentes techniques d'enregistrement qui peuvent être utilisées pour réaliser une ICM sont décrites dans la section 3 de ce chapitre page 22.

Pour les ICM non invasives, le signal enregistré contient peu d'informations et est d'un rapport signal sur bruit extrêmement faible. Cela fait de l'acquisition du signal le facteur le plus limitant pour les ICM et malheureusement on ne peut que se borner à attendre une nouvelle avancée technologique permettant des mesures plus précises et moins bruitées de l'activité cérébrale.

En plus d'être limitant pour le fonctionnement des ICM, le caractère expérimental (au sens propre du terme) des ICM est extrêmement coûteux pour la recherche. Le fait de concevoir, puis multiplier les expériences avec des sujets, prend énormément de temps.

On peut séparer la recherche qui est effectuée sur les ICM en deux catégories :

- **Les analyses hors-ligne (ou *offline*)** Certaines études ou améliorations peuvent se faire sur des données déjà enregistrées. Ceci permet de comparer rapidement et pour un faible coût de nombreuses méthodes (par exemple [Fruitet *et al.*, 2010]). La taille des bases de données utilisées et les méthodes de validation de type "validation croisée" ou "bootstrap" permettent d'obtenir des résultats statistiquement plus fiables qu'effectuer un nombre forcément limité d'expériences en-ligne. Le chapitre VI est consacré à une boîte à outils Matlab permettant l'étude et l'optimisation des ICM hors-ligne. Cependant, la validité des résultats ainsi obtenus doit être ensuite vérifiée par des expériences en-ligne.
- **Les analyses en-ligne (ou *online*)** Il s'agit de réaliser des expériences avec un traitement des données en temps réel. Ce type d'analyse est beaucoup plus coûteux que des analyses hors-ligne mais c'est la seule façon de garantir

le bon fonctionnement d'une ICM. Une description des expériences que nous avons réalisées se trouve dans le chapitre III.

Une partie du temps de ma thèse a été consacrée à mettre en place un système permettant de faire des expériences EEG puis à réaliser des acquisitions de données⁴.

1.2.2 Extraction du signal de contrôle

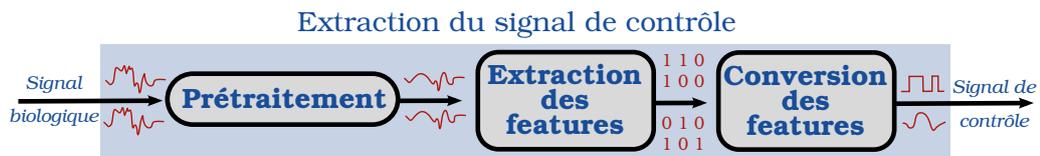


Fig. I.2 – Schéma de l'extraction du signal de contrôle à partir du signal biologique. Le signal de contrôle doit ensuite être transformé en commandes.

On dispose d'un signal numérique décrivant l'activité cérébrale duquel on veut extraire un signal de contrôle (dans la terminologie établie par Mason cette extraction s'appelle la transduction [Mason *et al.*, 2007]).

En pratique, outre le fait que le signal de contrôle doit être décodé du signal biologique, une grande difficulté provient du fait que les données qui nous intéressent sont généralement noyées dans le bruit.

Bien que le type de signaux et le genre d'ICM puissent être très variés, la conversion des signaux numériques provenant du cortex en signal de commande exploitable, est généralement composée des mêmes étapes. Ces étapes sont représentées sur le schéma I.2 et brièvement détaillées ci-dessous.

1. **Prétraitement** Le but du prétraitement est d'augmenter le rapport signal sur bruit et d'éliminer les artefacts.

Dans le cas de l'EEG, on commence généralement par appliquer un filtre passe-bande pour supprimer les fréquences qui ne nous intéressent pas et qui contiennent principalement du bruit (le courant 220V servant à alimenter les appareils produit par exemple d'importants artefacts à 50Hz). On peut ensuite appliquer un filtre spatial de type Laplacien (c'est à dire que l'on retire à chaque électrode la moyenne des potentiels mesurés sur les électrodes voisines) pour essayer d'augmenter

⁴Sous cette simple phrase se cachent des centaines d'heures passées à développer un driver linux pour le serveur d'acquisition et à enrichir OpenViBE (le logiciel utilisé pour réaliser les ICM) tout en luttant contre une source intarissable de bugs, puis à effectuer des enregistrements EEG.

la résolution spatiale de l'EEG. Il est aussi possible d'utiliser des techniques plus complexes pour retirer les artéfacts créés par des clignements d'yeux ou les contractions musculaires [[Semlitsch et al., 1986](#), [Bayliss et Ballard, 2000](#), [Shoker et al., 2005](#)].

2. **Extraction des features** Il s'agit d'extraire du signal des caractéristiques qui vont permettre de décider quel signal de contrôle doit être transmis. Il est courant d'utiliser la puissance extraite à certaines bandes de fréquences pour des électrodes spécifiques (c'est le cas pour les ICM basées sur les rythmes sensori-moteurs, section 4.2.2, ou le SSVEP, section 4.1.2).
3. **Conversion des features.** La conversion des features en signal de contrôle se fait généralement à l'aide d'algorithmes d'apprentissage supervisé. Il s'agit d'algorithmes de classification quand le signal de contrôle prend un nombre fini de valeurs, ou de régression quand l'ensemble de sortie est continu. Il faut dans un premier temps réaliser une base d'apprentissage qui va permettre à l'algorithme d'associer aux exemples (décrits par les valeurs des différentes features) le signal de contrôle désiré. Une fois l'apprentissage réalisé, l'algorithme va tenter de généraliser à tout nouvel exemple le contrôle le plus approprié. La section 3 du chapitre VI introduit une formalisation des algorithmes de classification puis décrit les différentes méthodes qui peuvent être utilisées pour évaluer leur performance.

Les différentes chaînes de traitement que nous avons utilisées sont présentées dans le chapitre III.

1.2.3 Conversion du signal de contrôle en commandes

Bien que le signal de contrôle puisse être utilisé directement il est préférable de le convertir en commandes efficaces et optimisées. Actuellement le signal de contrôle obtenu à la sortie du transducteur est peu précis, lent et possède un nombre de degrés de liberté limité. Pour rendre une ICM réellement utile il est donc souvent nécessaire de la rendre plus fiable, plus rapide ou plus complexe.

Exemple

Un linuxien écrivant sa thèse avec gVim veut la sauvegarder. Il devra taper sur 5 touches "esc : w q entrer". Clairement côté optimisation on aurait pu faire mieux, mais ce n'est finalement pas si grave, car il mettra moins d'une fraction de seconde à exécuter la commande. Maintenant s'il utilisait une ICM nécessitant 7 secondes [[Donchin et al., 2000](#)] pour produire chaque caractère, on comprendra que l'optimisation de la gestion des commandes n'est pas à négliger.

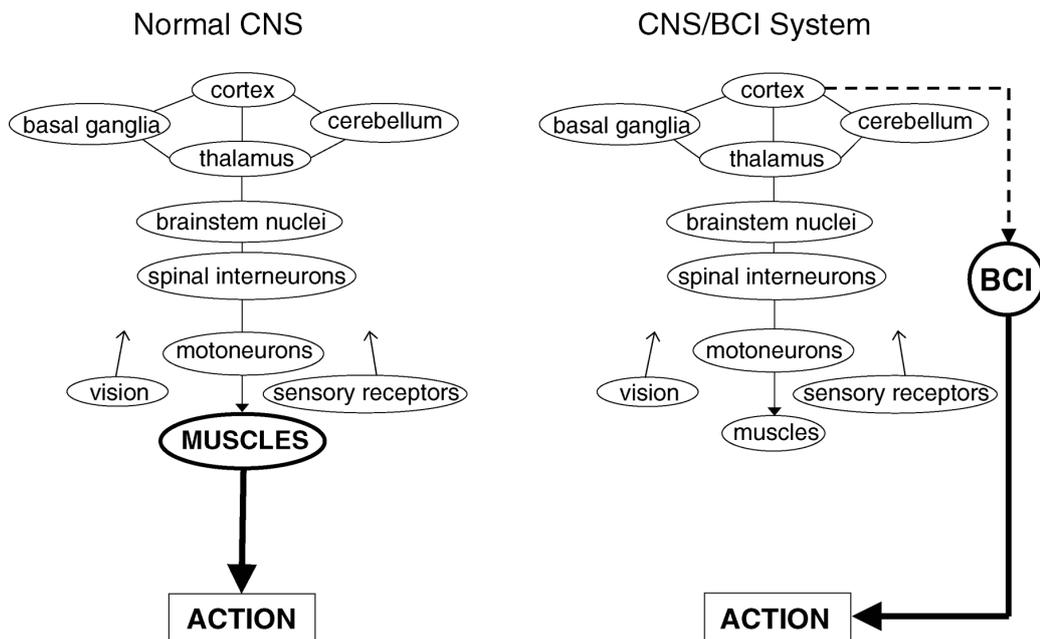


Fig. I.3 – Comparaison de la production d'un mouvement moteur et de l'utilisation d'une ICM. Le digramme de gauche représente les différents aires du SNC impliquées dans la réalisation d'une action motrice (cortex, ganglions de la base, cervelet, thalamus, tronc cérébral). L'utilisation d'une ICM est représentée à droite. L'ICM fait alors jouer aux neurones corticaux le rôle de neurones moteurs. Image reprise de [Wolpaw, 2007].

Les limites du signal de contrôle ont deux sources :

- l’acquisition du signal (contraintes technologiques) mais aussi la capacité du transducteur (celle-ci étant repoussée continuellement par la recherche sur les ICM),
- une limite intrinsèque aux ICM telles qu’elles sont conçues aujourd’hui : le cortex n’ayant pas évolué pour communiquer directement avec un ordinateur, les “commandes” contenues dans le signal ne sont pas directement exploitables et doivent être, une fois décryptées, post-traitées. Ce travail est normalement effectué par d’autres régions du système nerveux y compris la moelle épinière et le système nerveux périphérique (voir figure I.3) [Wolpaw, 2007].

À l’heure actuelle le travail effectué sur la conversion du signal de contrôle en commande reste marginale dans la recherche sur les ICM. De même que les ICM ont grandement bénéficié de connaissances venant du traitement du signal et de l’apprentissage, il est probable qu’un nouveau pas sera franchi quand des chercheurs en intelligence artificielle, robotique ou interfaces homme-machine, s’intéresseront au problème de la conversion du signal de contrôle en commandes.

Il ne s’agit pas simplement d’ajouter un module d’auto-complétion ou un dérivé du T9 au P300Speller [Vaughan *et al.*, 2006], mais réellement d’une nouvelle problématique de recherche à part entière. Les ICM sont des interfaces d’un type nouveau, dont la pauvreté du signal de contrôle ne fait que renforcer le besoin d’une excellente gestion des commandes.

1.2.4 *Le Feedback*

Une ICM est une boucle qui est refermée par l’envoi d’un feedback à l’utilisateur. Celui-ci est généralement visuel mais peut aussi être auditif ou somesthésique. Il peut être aussi simple que l’apparition d’une croix rouge ou d’un smiley pour signaler les mauvaises et bonnes commandes mais peut être bien plus complexe comme la navigation dans un environnement immersif. Pour de nombreux types d’ICM, l’utilisateur doit passer par une phase d’apprentissage pour réussir à contrôler convenablement l’interface. Cette apprentissage ne pouvant s’effectuer sans un retour vers l’utilisateur, la gestion du feedback n’est pas à négliger lors de la création d’une ICM [McFarland *et al.*, 1998].

Le feedback doit être clair et en rapport avec la tâche demandée tout en n’entrant pas en interaction avec la production des commandes. D’après [Leeb *et al.*, 2007] l’utilisation d’un environnement stimulant, comme la navigation spatiale dans un appartement virtuel, permet d’accroître les performances de l’ICM.

1.3 Terminologie sur les différents types d'ICM

Il existe une très grande variété d'ICM que l'on peut classer de différentes façons.

On peut opérer une première distinction suivant le type de méthode d'enregistrement utilisée. On parlera ainsi :

- d'ICM **invasives** nécessitant l'implantation d'électrodes.
- d'ICM **non invasives** utilisant des méthodes d'enregistrement non invasives.

Une autre classification dépend de la nature des signaux utilisés pour contrôler l'ICM. D'après la terminologie employée par S.G. Mason [[Mason et al., 2006](#), [Mason et al., 2007](#)], on distingue les ICM endogènes et les ICM exogènes :

- pour les ICM **endogènes**, les signaux utilisés sont spontanément générés par l'utilisateur, que ce soit de façon consciente ou non.
- pour les ICM **exogènes**, les signaux utilisés sont des réponses à des stimuli (généralement des potentiels évoqués comme le P300).

Il est aussi possible d'établir des distinctions sur le fonctionnement du protocole. On parlera de :

- protocoles **synchrones** quand on évalue la réponse de l'utilisateur à des instants précis.
- protocoles **asynchrones** quand l'utilisateur peut envoyer des commandes quand il le désire.

Le terme *self-paced* est quant à lui employé pour désigner des ICM que l'utilisateur peut activer et désactiver de lui-même⁵.

Il en découle qu'une ICM exogène ne peut généralement être utilisée qu'avec un protocole synchrone⁶. Par contre une ICM utilisant des signaux endogènes pourra être utilisée de façon synchrone ou asynchrone.

2 QUELQUES NOTIONS DE NEUROPHYSIOLOGIE

Le but de cette section est de donner une idée de l'origine du signal électromagnétique que l'on enregistre en EEG.

2.1 Fonctionnement d'un neurone

Le système nerveux central est constitué d'environ 100 milliards de neurones connectés entre eux par 10^{15} synapses et supportés par un nombre encore plus

⁵Il arrive cependant que le terme *self-paced* soit employé comme synonyme d'asynchrone.

⁶On peut imaginer comme exception une utilisation asynchrone du SSVEP.

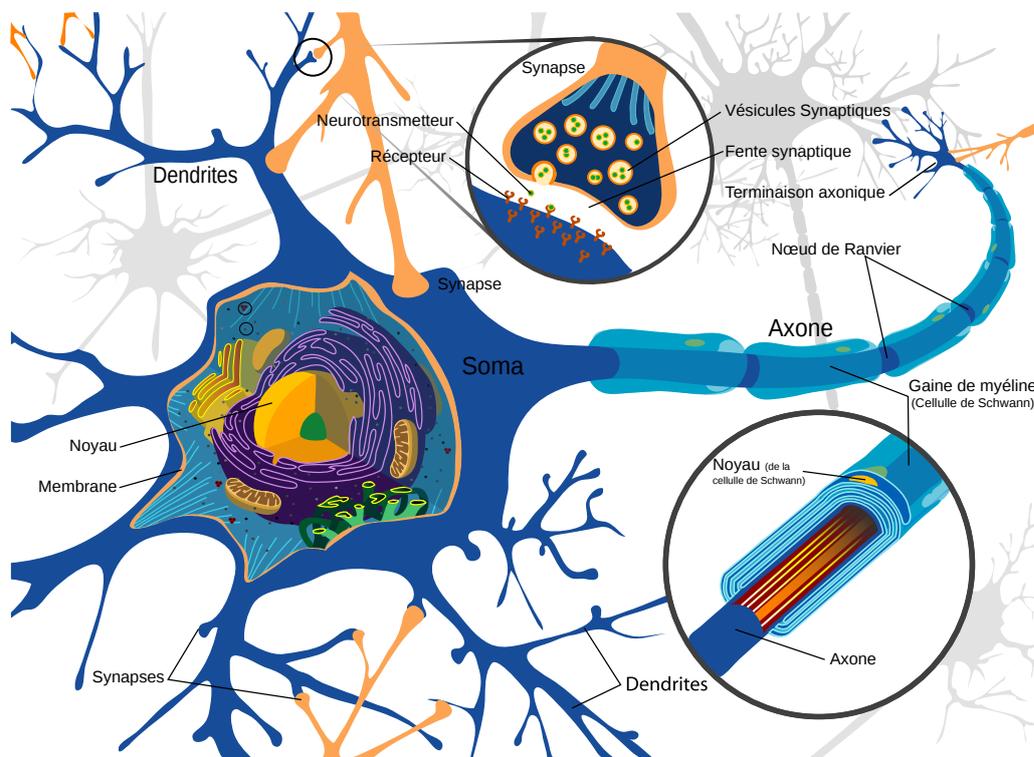


Fig. I.4 – Schéma d'un neurone. Adapté de Wikipedia

grand de cellules gliales⁷.

Chaque neurone (voir figure I.4) reçoit, par l'intermédiaire des synapses disposées sur ses dendrites, des informations d'autres neurones. Il les intègre et les transmet à son tour par son axone. Cette transmission et intégration d'information sont le résultat de processus chimiques et électriques.

D'une façon extrêmement simplifiée on peut résumer le fonctionnement des neurones ainsi :

- Quand le potentiel électrique⁸ du soma (ou corps cellulaire) d'un neurone dépasse un certain seuil, un potentiel d'action est émis.
- Ce potentiel d'action (qui est une hausse locale du potentiel électrique) est transmis le long de l'axone et est ré-amplifié à chaque nœud de Ranvier.
- Quand il atteint les synapses présentes au bout de l'axone, il déclenche la libération de neurotransmetteurs.
- Les neurotransmetteurs traversent la fente synaptique et se fixent sur les récepteurs du neurone post-synaptique (dit afférent).
- Suivant le type de neurotransmetteurs, excitateurs ou inhibiteurs, cela active différents canaux ioniques qui vont faire augmenter ou diminuer le potentiel électrique.
- Ces variations électriques du potentiel se propagent le long des dendrites jusqu'au soma.
- L'ensemble des stimulations reçues sur les différentes synapses présentes sur les dendrites est donc intégré au cours du temps.
- Quand le potentiel électrique de ce neurone afférent atteint le seuil, un potentiel d'action est émis et ainsi de suite.

2.2 Que peut-on espérer enregistrer de l'activité neuronale ?

Le potentiel d'action qui se propage le long de l'axone est d'une amplitude significative (environ 100mV) mais d'une durée très brève. Le champ électromagnétique qui en résulte est extrêmement faible et il faut donc placer des électrodes à proximité des neurones si l'on veut pouvoir les enregistrer.

À l'opposé, les potentiels postsynaptiques (excitateurs et inhibiteurs) sont d'amplitude moindre, mais ayant une dynamique beaucoup plus lente, engendrent des courants ioniques dans l'espace extracellulaire.

Il se trouve qu'une partie des neurones du cortex (les cellules pyramidales, voir figure I.5) disposent d'un grand dendrite orienté perpendiculairement à la surface du cortex. Les dendrites de ces neurones sont donc alignés, ce qui fait que les courants ioniques provoqués par les potentiels postsynaptiques se somment.

⁷Ces chiffres sont donnés de façon indicative, les variations étant importantes suivant les sources.

⁸Il s'agit de la différence de potentiel entre le milieu intra-cellulaire du neurone et le milieu extra-cellulaire.

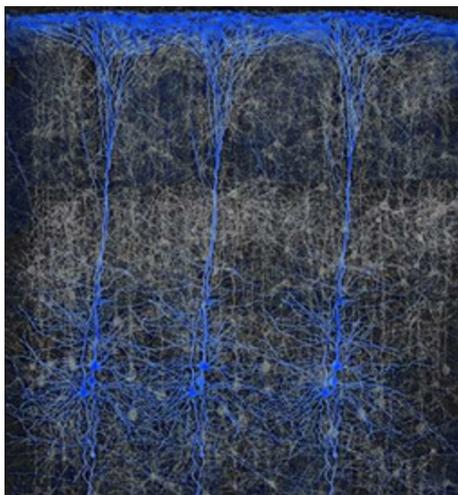


Fig. I.5 – Cellules pyramidales dans le néocortex. <http://brainmind.com>

L'activation d'une colonne fonctionnelle, un groupe d'environ 100 000 neurones réagissant ensemble à un stimulus, engendre ainsi des courants suffisamment forts pour être détectables depuis l'extérieur du crâne.

3 LES MÉTHODES D'ACQUISITION

De nombreuses techniques d'enregistrement permettent de capter (directement ou indirectement) l'activité cérébrale et de la convertir en signaux qui peuvent être utilisés pour réaliser une ICM.

Certaines de ces techniques, dites invasives, nécessitent l'implantation chirurgicale d'électrodes. Même si cette thèse ne porte absolument pas sur leur utilisation, la prochaine section (3.1) leur sera consacrée. D'autres techniques, dites non-invasives, se contentent de mesures depuis l'extérieur de la tête et ne comportent donc ni danger ni effet secondaire pour l'utilisateur. Ces techniques seront détaillées dans la section suivante (3.2).

3.1 Méthodes d'acquisition invasives

Elles sont principalement utilisées chez l'animal et consistent à implanter des électrodes sous le crâne, voire dans le cortex, pour mesurer le champ électrique résultant de l'activité neuronale. Bien que ces méthodes permettent d'enregistrer des signaux de meilleure qualité, elles ont de nombreux inconvénients liés à l'implantation chirurgicale des électrodes. Il peut y avoir des problèmes d'infection ou de lésions dues au mouvement des électrodes au cours du temps. Ces méthodes

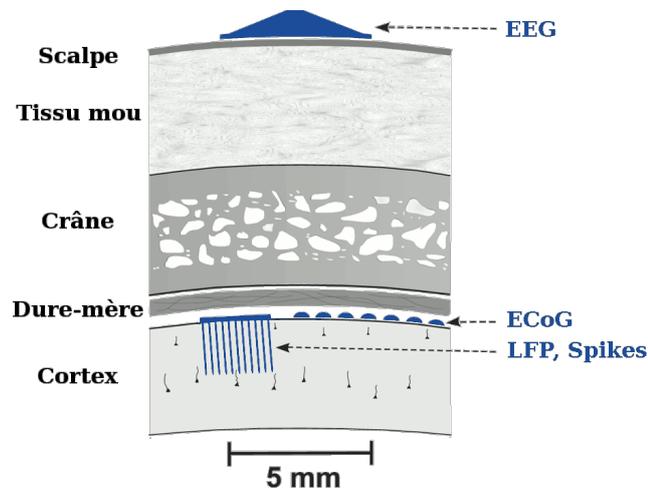


Fig. I.6 – Les différentes méthodes permettant de mesurer le champ électrique produit par le cerveau. EEG : électrodes posées sur le cuir chevelu (non invasif), ECoG : électrodes posées sur le cortex (invasif), LFP et Spikes : électrodes implantées dans le cortex (très invasif). Image adaptée de [Wolpaw et al., 2006]

posent aussi de nombreux problèmes éthiques que se soit pour leur utilisation chez l'homme ou chez l'animal.

Voici une brève description des principales techniques d'enregistrement invasives qui sont utilisées pour réaliser des ICM :

3.1.1 L'Électrocorticographie ou ECoG

Il s'agit de placer une matrice d'électrodes sur la dure-mère ou directement à la surface du cortex cérébral (voir figure I.6 et I.7). Les électrodes ne pénètrent donc pas dans le cortex. On enregistre des signaux électriques de même nature qu'en électroencéphalographie (ou EEG : électrodes posées sur le cuir chevelu). Cependant, le champ électrique n'ayant pas à traverser le crâne, les signaux sont de meilleure qualité : meilleur rapport signal sur bruit, meilleure précision spatiale et temporelle (en particulier pour les hautes fréquences). Par conséquent l'ECoG permet de distinguer des variations du champ électrique dues à des populations neuronales bien plus petites qu'en EEG.

3.1.2 Électrodes implantées dans le cortex

Certaines électrodes sont prévues pour être implantées dans le cortex (voir figure I.6 et I.8). Elles peuvent alors enregistrer les champs électriques locaux (Local Field Potentials ou LFP) provenant des activités synaptiques des neurones environnants ou même, pour les plus fines d'entre elles, enregistrer les potentiels d'action (ou *spikes*) émis par un ou plusieurs neurones. On parle alors d'électrodes

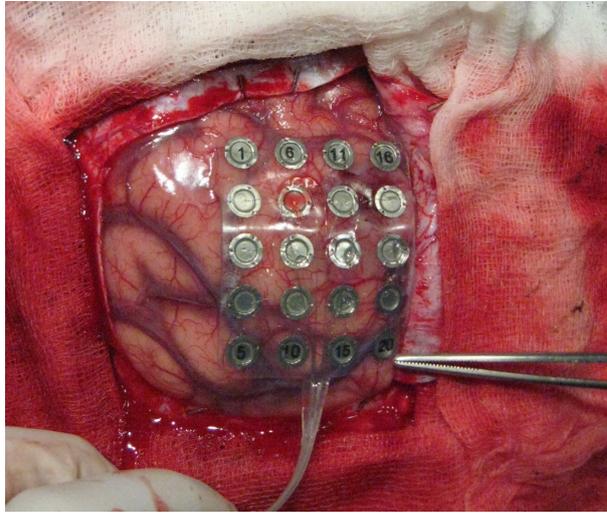


Fig. I.7 – Électrocorticographie ou ECoG. Implantation d'électrodes à la surface du cortex. Image reprise de <http://www.neuromonitoreo.8m.net/>

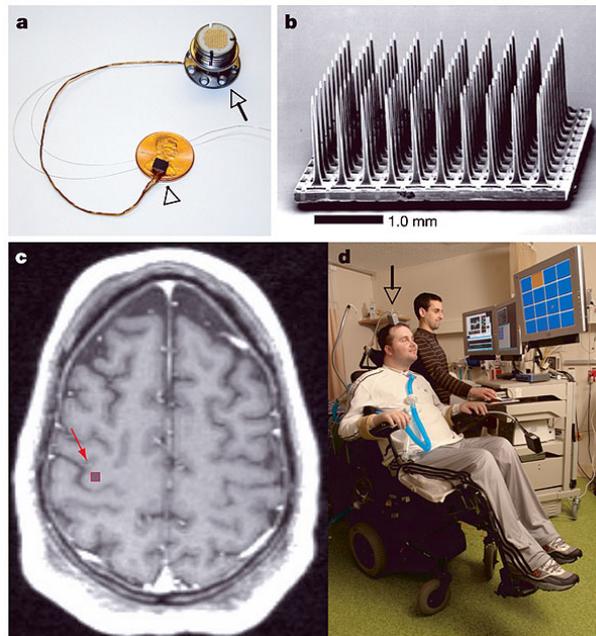


Fig. I.8 – Un exemple d'électrodes implantées. **a** : Le capteur BrainGate, posé sur un penny. **b** : les 100 électrodes du capteur. Elles mesurent 1 mm de long et sont espacées de 0.4 mm. **c** : IRM du sujet, la flèche indique le lieu de l'implantation. **d** : Le premier participant de l'expérience BrainGate. Il regarde l'écran et doit diriger le curseur vers la case orange. Image reprise de [Hochberg et al., 2006]

unitaires ou multi-unitaires.

De telles méthodes d'enregistrement permettent d'obtenir des signaux d'une extrême précision mais posent des problèmes car elles sont extrêmement invasives. En particulier, la qualité des signaux peut baisser dans les mois qui suivent l'implantation suite à une réaction de l'organisme face à la présence d'un corps étranger (phénomène de fibrose).

Exemple : BrainTV

L'utilisation d'électrodes implantées est courante pour le diagnostic pré-chirurgical de l'épilepsie (sEEG). BrainTV permet à des patients ayant des électrodes implantées dans le cortex de voir en temps réel l'activité de certaines zones de leur cerveau [Lachaux et al., 2007].

3.2 Méthodes d'acquisition non invasives

3.2.1 Électroencéphalographie ou EEG



Fig. I.9 – Mise en place du casque EEG dans le laboratoire de l'équipe.

On mesure le potentiel électrique à l'aide d'électrodes placées sur le scalp, voir figure I.9. Un contact électrique est généralement établi grâce à l'adjonction d'un gel conducteur. Cette méthode a comme avantage d'être relativement peu coûteuse, d'être transportable et d'avoir une bonne résolution temporelle. Cela en fait la méthode la plus utilisée pour réaliser des ICM chez l'homme et c'est cette méthode qui a été utilisée pour les expériences réalisées durant cette thèse.

Elle a cependant trois principales limites provenant du fait que les signaux électriques enregistrés ont dû traverser le crâne :

- Une résolution spatiale très limitée : on ne peut pas distinguer des signaux ayant des origines voisines dans le cortex (ceci est lié au crâne qui agit comme un filtre “passe-bas” spatial).

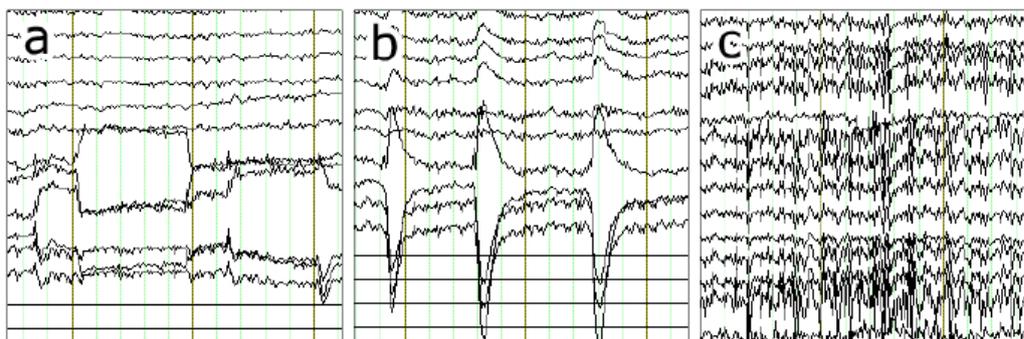


Fig. I.10 – Artefacts pouvant être présent dans les signaux EEG. A : mouvements oculaires. B : clignements des yeux. C : tension musculaire.

- Un rapport signal sur bruit très faible. Par exemple, les clignements d’yeux créent des artefacts bien plus puissants que les signaux utilisés pour les ICM (voir figure I.10).
- Une perte de puissance dans les hautes fréquences. Les fréquences supérieures à une quarantaine de Hertz sont extrêmement atténuées⁹ [Nunez, 2005].

Le chapitre V de ce manuscrit, est consacré à l’amélioration de la résolution spatiale de l’EEG pour les ICM par reconstruction de l’activité des sources corticales. D’après [Vion-Dury et Blanquet, 2008], l’utilisation direct des électrodes de l’EEG permet qu’une analyse au niveau des aires de Brodmann (soit des populations d’environ 10^8 neurones répartis sur des colonnes de 50mm de diamètre). La reconstruction de l’activité corticale par résolution du problème inverse permet d’obtenir une résolution spatiale de l’ordre du cm [Leahy et al., 1998, Grech et al., 2008], ce qui reste cependant légèrement supérieur à la taille des macro colonnes (de 0.5 à 3mm de diamètre comprenant de 10^5 à 10^6 neurones).

Un autre inconvénient provient du temps nécessaire à la mise en place des électrodes. Il faut compter une quinzaine de minutes pour un système ne disposant que de quelques électrodes et jusqu’à quarante minutes voire plus d’une heure pour les systèmes disposant de 64 à 256 électrodes. Ce temps, déjà contraignant quand on réalise des expériences, peut être une importante barrière à une utilisation quotidienne par des patients.



Fig. I.11 – La MEG de l'INSERM de Lyon.

3.2.2 Magnétoencéphalographie ou MEG

La MEG (figure I.11) est d'un principe similaire à l'EEG si ce n'est que l'on mesure le champ magnétique et non le potentiel électrique [Hämäläinen *et al.*, 1993]. Elle permet théoriquement une meilleure précision spatiale [Leahy *et al.*, 1998], le champ magnétique étant moins sensible à la traversée du crâne.

Cependant le champ magnétique émis à l'extérieur du crâne étant extrêmement faible (de l'ordre de quelques femto-Teslas, soit $\sim 10^{-14}T$ à comparer au $10^{-4}T$ du champ magnétique terrestre) cela génère de fortes contraintes :

- les mesures doivent être réalisées dans une chambre isolée magnétiquement¹⁰.
- les capteurs sont des magnétomètres à SQUID qui doivent être maintenus à $4K$ et donc refroidis à hélium liquide.

Pour ces raisons la MEG n'est pas transportable et est bien plus coûteuse que l'EEG. C'est donc une technique intéressante pour la recherche en ICM mais qui ne pourra pas être sortie du laboratoire dans un avenir proche.

⁹À cause du filtrage passe-bas spatial du crâne, l'EEG moyenne des population spatialement étendues. Or plus les fréquences d'oscillation sont importantes et plus la population est étendue moins la synchronisation est probable.

¹⁰Le blindage nécessaire n'a rien de commun (et est extrêmement plus coûteux) avec la cage de Faraday dans laquelle on effectue généralement les mesures EEG.

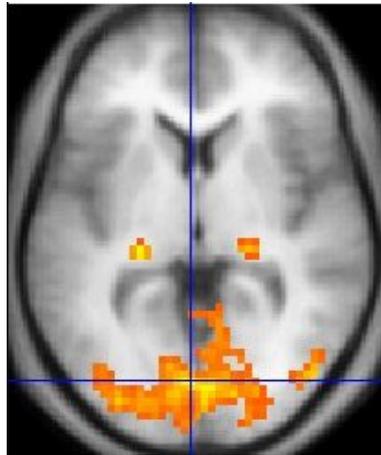


Fig. I.12 – Détection par l'IRMF de l'activation des régions du cerveau impliquées dans la perception visuelle. Image reprise de [wikipedia](#).

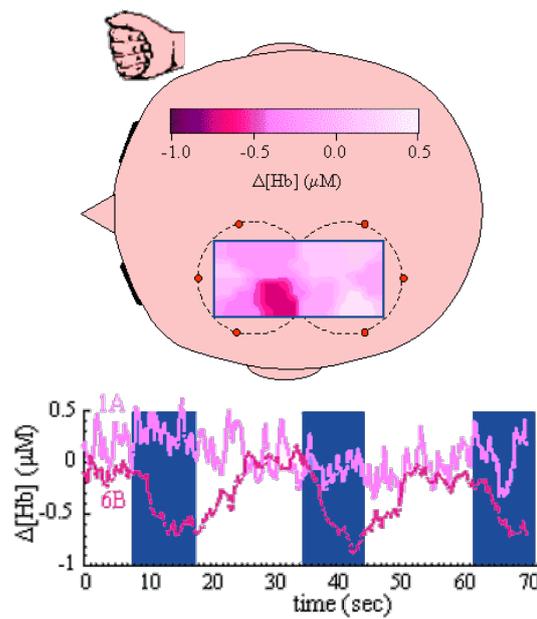


Fig. I.13 – Exemple de mesures établies par imagerie proche infra-rouge. Le sujet bouge la main droite durant des périodes de 10 secondes représentées par des barres bleues. Les deux tracés représentent l'évolution de la concentration de deoxy-hémoglobine en deux points. Une seconde environ après chaque période de mouvement de la main droite la courbe rouge diminue et remonte progressivement après la période de mouvement. Image reprise de [Franceschini et al., 2000]; Un gif animé est disponible à cette adresse : <http://www.opticsinfobase.org/oe/oeimage/655.gif>

3.2.3 *Imagerie par résonance magnétique fonctionnel ou IRMf*

Contrairement aux techniques d'enregistrement décrites jusqu'à présent l'IRMf donne une mesure indirecte de l'activité cérébrale (figure I.12 et I.19) basée sur l'effet BOLD (blood-oxygen-level dependent). En quelques mots : l'augmentation de l'activité neuronale engendre une plus grande consommation d'oxygène qui est surcompensée par le système vasculaire. L'hémoglobine n'ayant pas les mêmes propriétés magnétiques quand elle est liée au dioxygène, c'est l'augmentation du flux sanguin qui est mesurée en IRMf, reflétant indirectement l'activité cérébrale. Il en résulte que l'IRMf a une très bonne résolution spatiale, de l'ordre du millimètre, mais une résolution temporelle¹¹ supérieure à la seconde qui la rend difficilement utilisable pour des traitements en temps réel et à fortiori pour développer des ICM. Comme la MEG, l'IRMf est très coûteux et non transportable. L'IRMf a cependant été utilisé pour réaliser plusieurs ICM [Weiskopf *et al.*, 2004, Yoo *et al.*, 2004].

3.2.4 *Imagerie proche infra-rouge (near infrared spectroscopy ou NIRS)*

C'est une technique extrêmement récente. Elle repose, comme l'IRMf, sur l'effet BOLD mais cette fois le taux d'hémoglobine oxygénée est mesuré à l'aide de caméras infra-rouge. Elle a l'avantage, par rapport à l'IRMf, d'être abordable et relativement transportable. De plus la fréquence d'échantillonnage est significativement plus élevée (environ 100Hz) mais l'activité neuronale étant indirectement mesurée par l'effet BOLD, la résolution temporelle reste insuffisante pour effectuer des traitements véritablement en temps réel (il y a un retard d'environ 1 seconde, voir figure I.13). Son utilisation dans le cadre des ICM reste encore anecdotique [Nagaoka *et al.*, 2010].

3.3 Pourquoi avons-nous choisi l'EEG ?

À l'heure actuelle, si l'on cherche une méthode d'acquisition non invasive qui soit portable et qui ait une résolution temporelle suffisante pour faire du traitement en temps réel il n'y a que l'EEG (voir figure I.14).

Il faut cependant garder en mémoire qu'une partie des utilisateurs potentiels d'ICM souffrent de problèmes moteurs graves. Utiliser des méthodes invasives n'est alors pas forcément à proscrire, si elles permettent un véritable gain par rapport à des ICM utilisant l'EEG.

¹¹L'acquisition d'une image volumique du cerveau durant environ 1 seconde, la fréquence d'échantillonnage de l'IRMf dépasse à peine le Hertz quand on travaille sur des zones réduites du cerveau.

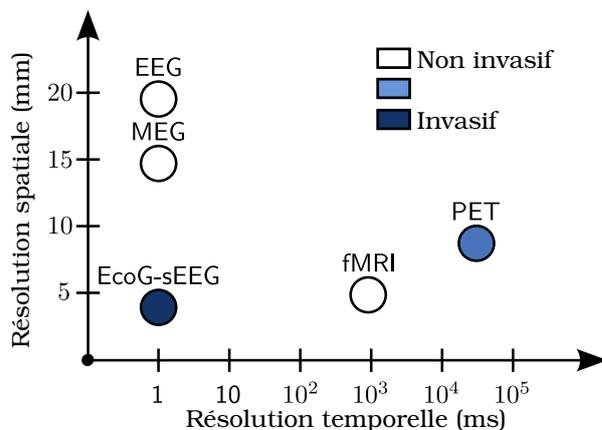


Fig. I.14 – Résolution spatiale et temporelle des différentes méthodes d'enregistrement.

4 LES ICM UTILISANT L'EEG

Le but de cette section est de montrer la diversité des informations issues de l'EEG qui peuvent être exploitées pour réaliser des ICM.

La première partie est consacrée aux ICM exogènes. Les commandes de l'utilisateur sont dans ce cas des réponses du cortex à des stimuli sensoriels, appelés potentiels évoqués (ou ERP pour *event related potential*). Il s'agit de signaux caractéristiques se produisant de façon automatique et non consciente en réponse à des stimuli précis.

La seconde partie est dédiée aux ICM endogènes. Les commandes sont alors spontanément générées par l'utilisateur. Les signaux peuvent être produits, ou plus généralement modulés, de façon consciente ou inconsciente par l'utilisateur. C'est sur ce type d'ICM que nous avons travaillé.

4.1 ICM basées sur les potentiels évoqués

Un potentiel évoqué est une variation du potentiel électrique mesurée dans le système nerveux en réponse à un stimulus [Duncan *et al.*, 2009]. Certains de ces potentiels évoqués, de par leur caractéristiques, permettent de réaliser des ICM.

4.1.1 ICM utilisant le P300

Le potentiel évoqué P300 C'est une variation positive du potentiel électrique qui a lieu en réponse à la survenue d'un stimulus rare et attendu [Farwell et Donchin, 1988]. Il a lieu principalement dans les aires pariétales

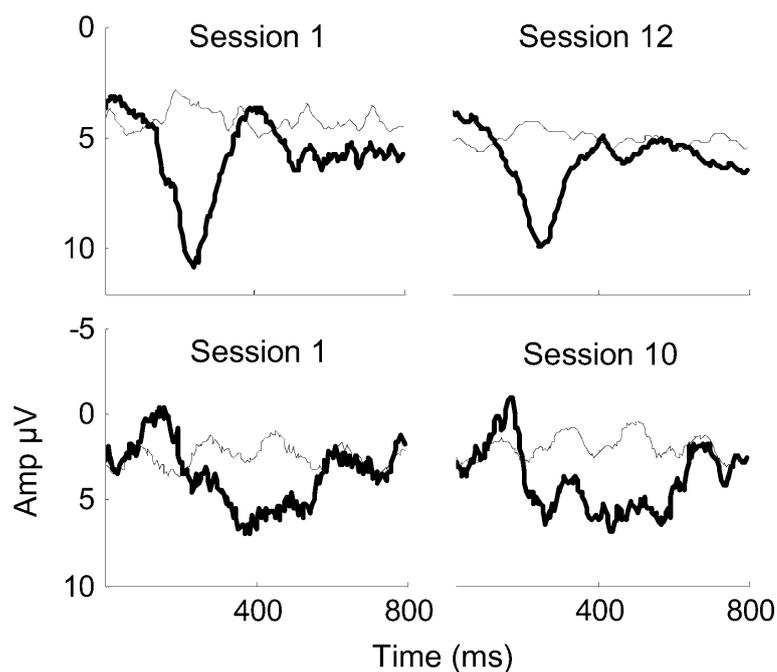


Fig. I.15 – Forme du P300 pour deux patients souffrant de sclérose latérale amyotrophique (chaque ligne représente des données d'un sujet). Pour que la forme caractéristique du P300 soit visible on moyenne sur un grand nombre d'essais (2040 essais pour le stimulus cible, courbe en gras, et 10200 essais pour les stimuli ne devant pas produire de P300, courbe en pointillés). Les courbes représentées sur la première ligne sont comparables à des P300 mesurés chez des sujets sains. Le pic positif du P300 à environ 300ms est bien visible (une convention veut que le positif soit représenté vers le bas sur les données électrophysiologiques). Pour l'autre sujet, la forme du P300 est moins typique. Image reprise de [Sellers et al., 2006a].

et environ 300 millisecondes (figure I.15) après le stimulus cible (d'où son nom : P pour positif et 300 pour le délai de 300 millisecondes)¹².

Il peut être facilement généré avec une expérience de type *oddball*. Le sujet est soumis à une série de stimuli (visuels ou auditifs) de deux types : un stimulus fréquent et un stimulus cible plus rare. Quand le stimulus cible a lieu au milieu des stimuli fréquents, cela déclenche un P300. Les P300 sont produits que le sujet soit conscient ou non d'attendre le stimulus cible, mais son occurrence doit rester suffisamment rare pour que le sujet soit "surpris" par son apparition.

Exemple du P300Speller C'est la première et la plus connue des ICM utilisant le P300 [Farwell et Donchin, 1988]. Il s'agit de présenter à l'utilisateur un clavier sous forme de matrice et de déterminer à l'aide de détection de P300 quelle lettre il veut sélectionner (figure I.16).

Le principe est simple : chaque ligne et colonne de la matrice est mise en surbrillance, ou flashée, dans un ordre aléatoire un certain nombre de fois. On demande à l'utilisateur de compter le nombre de fois où la lettre qu'il veut sélectionner est flashée.

Quant la colonne ou ligne qui contient la lettre cible est flashée, ce qui correspond à un stimulus rare et attendu, un P300 est généré. Il "suffit" alors de détecter les P300 et on en déduit que la colonne ou ligne qui a été flashée 300 millisecondes plus tôt contenait la lettre cible.

Le problème est que la détection des P300 n'est pas évidente. Il est en effet difficile de déterminer sur un seul essai si un P300 a eu lieu. Il faut généralement moyenner plusieurs essais (entre 2 et 10) et donc faire flasher chaque ligne et chaque colonne plusieurs fois.

La recherche sur le P300 Depuis 1988 de plus en plus d'équipes de recherche se sont mises à travailler sur l'utilisation du P300 dans les ICM, soit pour améliorer les taux de détection du P300 soit pour développer d'autres ICM sur le même principe [Sellers et Donchin, 2006]. Différents choix sont alors proposés à l'utilisateur tout en étant "flashés" (unitairement ou par groupes) dans un ordre aléatoire. La détection des P300 déclenchés par le stimulus cible permet de déterminer le choix de l'utilisateur.

Exemple

L'ICM Brain-Painting permet ainsi à des patients paralysés de peindre en sélectionnant pinceaux et points d'applications grâce à un P300Speller modifié [Munssinger et al., 2010] _____

On peut dégager différents axes de recherche liés au P300 :

- **L'amélioration de la détection des P300.** Améliorer la détection des P300 permet de rendre l'ICM plus fiable mais aussi plus rapide, puisqu'on va

¹²Plus précisément, le P300 est composé de deux composantes : le P3a et le P3b [Polich, 2007].

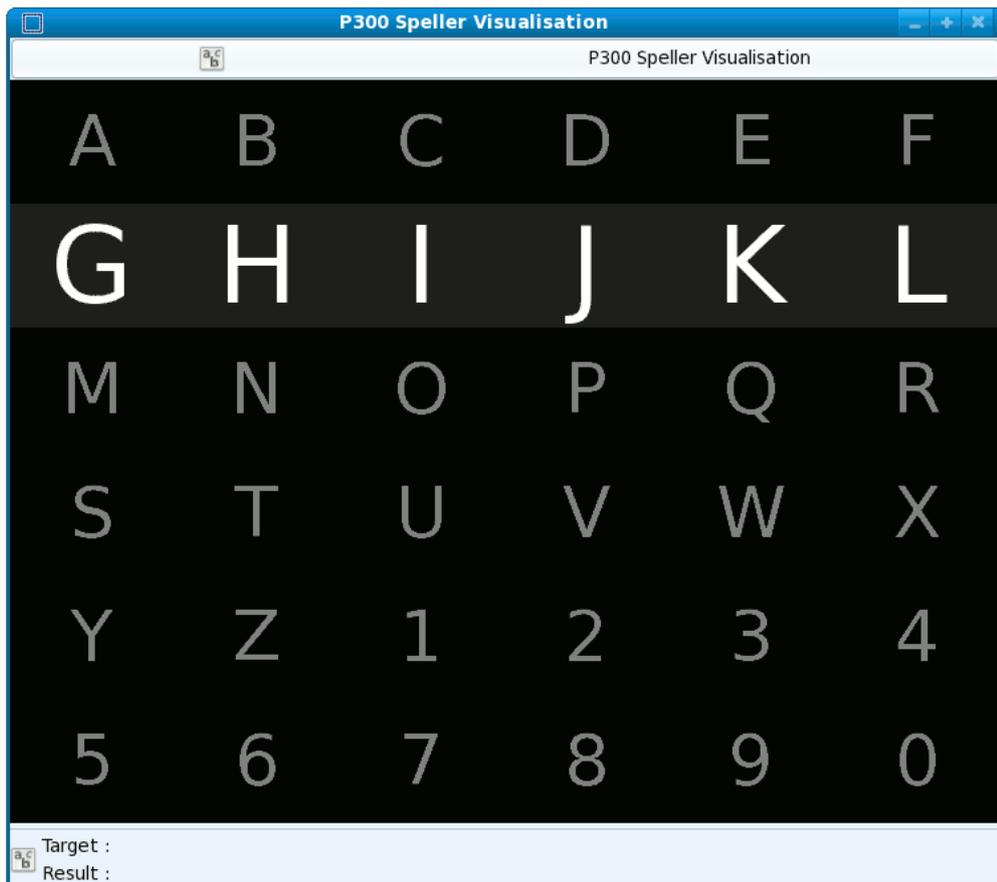


Fig. I.16 – La matrice de lettres du P300Speller utilisée dans OpenViBE. Il faut imaginer que la ligne mise en évidence reste blanche pendant seulement un dixième de seconde (on parle de flash). Si l'utilisateur souhaite écrire une lettre appartenant à la ligne flashée (donc ici de G à L) un P300 aura lieu environ 300 millisecondes après le flash, mais s'il se concentre sur une lettre d'une autre ligne, aucun P300 ne sera enregistré.

pouvoir diminuer le nombre d'essais que l'on doit moyenner pour détecter un P300. Il s'agit souvent de travailler sur les techniques de classification qui vont permettre de distinguer les essais contenant des P300 de ceux n'en contenant pas [Kaper *et al.*, 2004], mais d'importants progrès ont aussi été obtenus en travaillant sur le pré-processing ou sur l'extraction des caractéristiques (ou features). On notera par exemple l'utilisation de filtres spatiaux optimisés pour faire "ressortir" les P300 [Rivet *et al.*, 2009]

- **L'amélioration de l'interface de stimulation.** À cause de la stimulation en ligne et colonnes un nombre important d'erreurs sont localisées sur la même ligne ou colonne que la lettre cible est généralement proche de celle-ci. Plusieurs études explorent différentes méthodes de stimulation, en particulier par groupe pseudo-aléatoire de lettres [Townsend *et al.*, 2010]. Dans [Sellers *et al.*, 2006b, Takano *et al.*, 2009] l'impact du contraste et de la couleur de la matrice de stimulation sur les performances est évalué.
- **Améliorer la partie traitement de la commande de l'interface :** par exemple, dans le cas du P300Speller, utiliser un dictionnaire prédictif [Vaughan *et al.*, 2006]. Relativement peu de travail est effectué sur ce point, alors qu'il nous paraît potentiellement le plus bénéfique pour l'utilisateur.
- **Sortir le P300 du laboratoire.** Il s'agit d'abord de tester les ICM avec des sujets ayant des troubles moteurs dans les laboratoires [Sellers et Donchin, 2006], puis d'évoluer vers une utilisation à l'extérieur (environnement hospitalier ou domicile) [Vaughan *et al.*, 2006]. D'après [Brunner *et al.*, 2010, Treder et Blankertz, 2010] les performances du P300Speller sont grandement diminuées quand le sujet n'est pas capable de fixer la lettre cible et se contente d'y porter son attention.

Avantages et inconvénients des ICM basées sur le P300 Les ICM utilisant le P300 permettent, à l'heure actuelle, les meilleurs taux de transfert. L'autre point positif est qu'étant basées sur une réponse intrinsèque du système nerveux, elles ne nécessitent pas d'apprentissage de la part de l'utilisateur. En d'autres termes il n'a pas besoin d'apprendre à générer des P300. La prise en main¹³ d'un P300Speller est donc rapide, le contre-point étant qu'il n'y a pas ou très peu d'amélioration des performances liées à l'utilisation.

On peut également reprocher au P300 d'être fatigant visuellement. Les autres inconvénients sont intrinsèques à l'utilisation de potentiels évoqués : les commandes n'étant en réalité que des réponses de l'utilisateur à des stimuli, l'utilisateur n'a que très peu de liberté. Il ne peut influencer ni sur le timing des commandes, ni sur leur amplitude.

¹³hum... passons.

4.1.2 *Steady State Visual Evoked Potential (SSVEP)*

Le SSVEP est, comme le P300, un potentiel évoqué. Il est provoqué par la présentation d'un stimulus visuel ayant une fréquence de répétition supérieure à environ 3Hz (comme par exemple un carré clignotant du noir au blanc 10 fois par seconde). Le système nerveux soumis à un tel stimulus va générer dans les aires visuelles de l'activité électrique oscillant à la même fréquence (et/ou aux harmoniques). Ce qui est très intéressant pour les ICM, est que la puissance mesurée par EEG à la fréquence de stimulation augmente quand le sujet regarde directement le stimulus ou même porte simplement son attention dessus (figure I.17).

On peut alors réaliser une ICM en montrant simultanément à l'utilisateur plusieurs stimuli (clignotant à des fréquences différentes) illustrant plusieurs choix. L'utilisateur n'a alors qu'à porter son attention sur un des stimuli pour envoyer une commande. Un exemple d'ICM basée sur le SSVEP est illustré par la figure I.18.

Avantages et inconvénients des ICM basées sur le SSVEP Comme pour le P300, le SSVEP étant automatiquement produit, l'utilisateur n'a pas besoin d'effectuer d'apprentissage¹⁴. J'ai réalisé, durant un stage de M1, des expériences d'ICM utilisant les SSVEP et c'est donc en connaissance de cause que je peux affirmer que malheureusement les stimuli utilisés sont généralement très violents (encore plus que ceux faits pour générer des P300) et que l'utilisation de telles ICM est donc particulièrement fatigante.

¹⁴Un apprentissage est cependant nécessaire pour calibrer le transducteur.

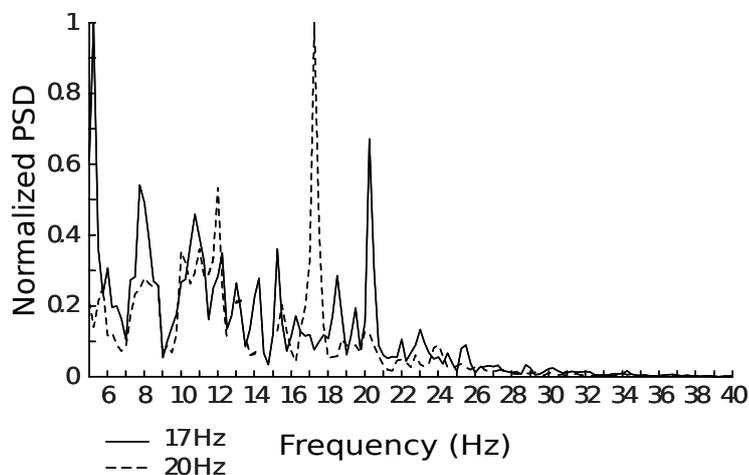


Fig. I.17 – Spectre de puissance mesuré en EEG quand l'utilisateur porte son attention sur un stimulus clignotant à 17Hz (trait en pointillés) ou 20Hz (trait plein). On remarque clairement la modulation de l'amplitude des pics à 17 et 20Hz en fonction de là où est portée l'attention du sujet. Image reprise de [Lalor et al., 2005].



Fig. I.18 – Exemple de BCI utilisant le SSVEP [Lalor et al., 2005]. Le personnage avance sur la corde et l'utilisateur doit l'aider à se stabiliser en portant son attention sur le damier clignotant situé du bon côté (dans le cas présent, celui de droite). Les deux damiers clignotent à des fréquences différentes. Quand l'utilisateur porte son attention sur l'un des damiers, la puissance à cette fréquence de stimulation augmente dans le cortex visuel, ce qui permet de déterminer quelle cible il veut sélectionner.

4.1.3 Potentiels d'erreur

Comme son nom l'indique, le potentiel d'erreur (ou PE) est une variation du potentiel électrique lié aux erreurs. Il peut soit être généré quand le sujet commet une erreur (qu'il en soit conscient ou non), soit quand il observe quelqu'un (ou un processus) en commettre une, soit encore quand il obtient un retour négatif sur le déroulement d'une tâche qu'il devait effectuer (figure I.19) [Ferrez et Millan, 2008, Ferrez et Millán, 2008, Lopez-Larraz *et al.*, 2010].

Contrairement au P300 ou au SSVEP, le potentiel d'erreur ne peut donc pas être utilisé pour réaliser une ICM au sens strict du terme, mais peut être exploité en sus de la plupart des ICM existantes afin de corriger les erreurs liées à l'utilisation de l'ICM.

Exemple d'utilisation du potentiel d'erreur pour le P300 Un sujet utilisant un P300Speller peut écrire environ une lettre toutes les 10 secondes avec un taux de réussite d'environ 95%. Il est donc très intéressant de pouvoir détecter les cas où la lettre sélectionnée par le speller n'est pas la bonne, voire de la corriger automatiquement [Dal Seno *et al.*, 2010].

Pour sélectionner une lettre, le sujet doit porter son attention sur une des lettres de la matrice du P300Speller, alors que les lignes et colonnes sont flashées aléatoirement. Une fois que chaque ligne et colonne a été flashée un certain nombre de fois (chaque flash de la lettre cible devant générer un P300), on peut évaluer la probabilité que chaque lettre soit celle voulue par l'utilisateur. On sélectionne la lettre la plus probable et le résultat est communiqué à l'utilisateur sous forme de feedback. Or si la lettre affichée n'est pas celle que l'utilisateur voulait sélectionner un potentiel d'erreur va être généré. Si on arrive à détecter ce potentiel d'erreur, on va pouvoir corriger la sélection, soit simplement en annulant la sélection de la lettre, soit en proposant la deuxième lettre la plus probable, voire la troisième si la présentation de cette nouvelle lettre génère aussi un potentiel d'erreur.

Avantages et inconvénients de l'utilisation du potentiel d'erreur Les avantages que permettrait une détection fiable des potentiels d'erreurs, que ce soit lors de l'utilisation d'une ICM ou d'une interface traditionnelle, sont nombreux. En effet, savoir que l'utilisateur vient de commettre une erreur peut permettre d'annuler la dernière commande voire de la corriger de façon automatique.

Le problème est que détecter les potentiels d'erreur en essai unique n'est pas facile et qu'il n'est pas possible de moyenniser les signaux sur plusieurs essais (il faudrait que tous les essais comportent une erreur ou qu'ils en soient tous dépourvus, ce qui perd tout de son intérêt).

Un autre inconvénient est que si le taux de faux positifs est trop élevé, utiliser le potentiel d'erreur peut faire chuter les performances. Reprenons l'exemple du P300Speller : si 1 fois sur 10 notre algorithme pense détecter un PE alors qu'il

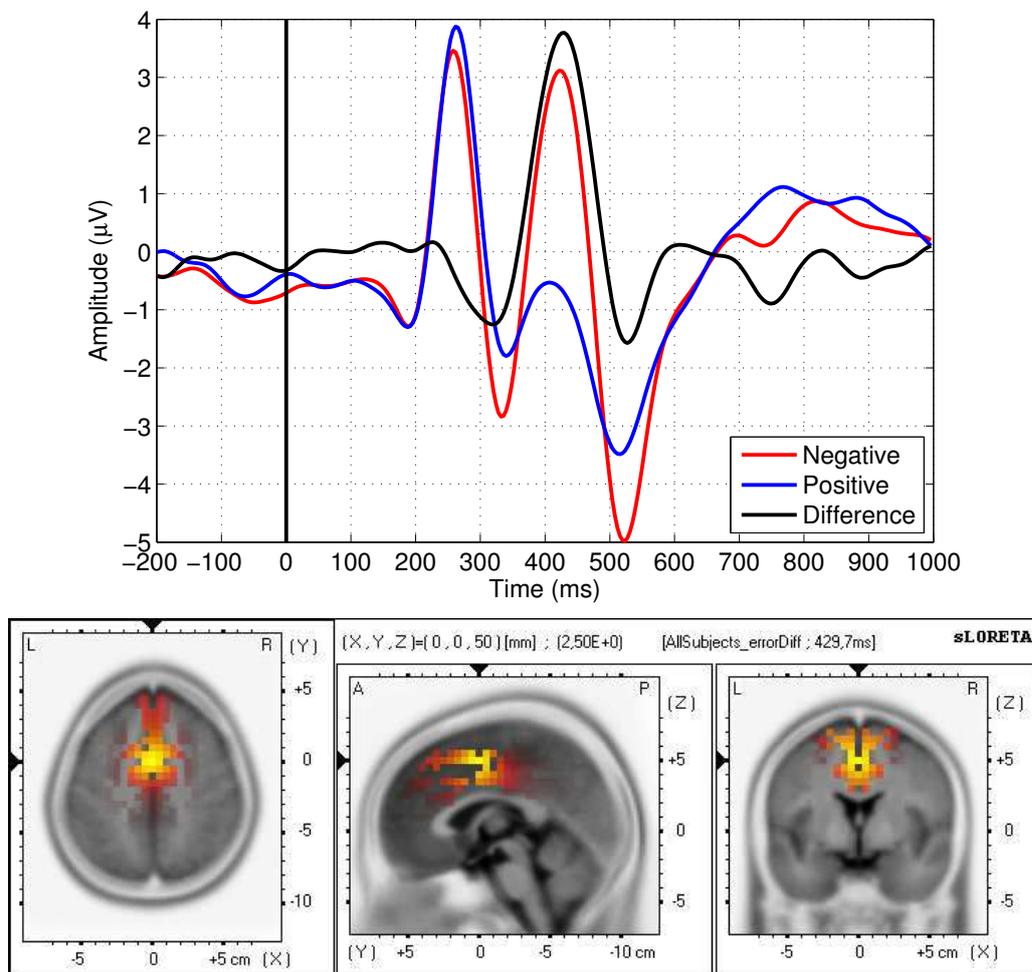


Fig. I.19 – Haut : Exemple de potentiel d'erreur. Les courbes proviennent d'une moyenne établie sur 5 sujets et un grand nombre d'essais. La courbe bleue correspond à l'activité mesurée sur l'électrode FCz après la présentation d'un feedback positif (l'utilisateur n'avait pas fait d'erreur) et la rouge à un feedback négatif (il y avait une erreur). La courbe noire est la différence des deux pour mettre en valeur le potentiel d'erreur. On voit clairement une différence environ 400ms après la présentation du feedback. Les P300 produits par la présentation du feedback sont aussi clairement visibles. Bas : Localisation des potentiels d'erreurs. Image reprise de [Lopez-Larraz et al., 2010].

n'y avait pas d'erreur, 10% des bonnes lettres sélectionnées seront corrigées par des erreurs, ce qui sera plus pénalisant que de ne pas détecter les 5% d'erreurs originelles.

4.2 ICM utilisant des signaux spontanément générés par l'utilisateur

L'inconvénient des ICM basées sur les potentiels évoqués est que l'utilisateur ne peut envoyer des commandes que sous forme de réponses à des stimuli. Il ne peut donc ni initier la communication, ni choisir les instants où il envoie les commandes. Heureusement, d'autres types de signaux peuvent être utilisés pour réaliser des ICM et s'affranchir de ces limites.

4.2.1 ICM basées sur la modulation des ondes corticales lentes, ou bio-feedback

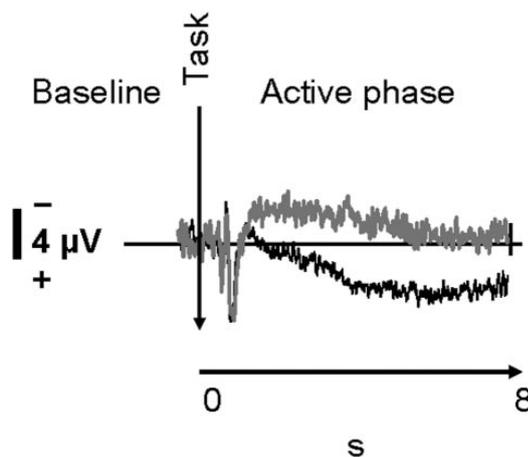


Fig. I.20 – Exemple de modulation d'ondes corticales lentes en fonction d'un stimulus de consigne (qui apparaît ici au temps noté 0). Les signaux sont moyennés sur 39 essais. On peut aussi voir le P300 déclenché par la consigne apparaitre avant la modulation des signaux. Image reprise de [Strehl et al., 2006]

Ce sont des ICM dont il est relativement aisé de comprendre le fonctionnement. On montre au sujet le signal EEG provenant d'une électrode et il doit se "concentrer" pour faire monter ou descendre le tracé. Le sujet apprend ainsi à contrôler les variations très basses fréquences (de l'ordre de 1Hz) de l'EEG simplement en regardant le signal [Elbert et al., 1980] [Birbaumer et al., 2000]. Il va ainsi pouvoir envoyer une commande en maintenant le niveau du signal mesuré en dessous d'un seuil pendant une certaine durée (voir figure I.20).

Apprendre à contrôler ses ondes corticales lentes permet donc d'envoyer spontanément des commandes à un ordinateur. Malheureusement les ICM basées sur ce principe souffrent de plusieurs limites :

- L'apprentissage que doit effectuer l'utilisateur pour contrôler l'interface est très long.
- Plusieurs secondes sont requises pour ne transmettre qu'un bit. Les taux de transfert d'information sont donc faibles.
- Il n'est pas possible d'augmenter le nombre de dimensions de contrôle.

On notera également qu'apprendre à contrôler ses ondes corticales peut être utilisé à des fins thérapeutiques (par exemple chez des enfants hyperactifs [Strehl *et al.*, 2006]).

4.2.2 *Imagination motrice et rythmes sensori-moteurs*

Le travail décrit dans ce manuscrit porte exclusivement sur ce type d'ICM. Un état de l'art des ICM utilisant les rythmes sensori-moteurs est effectué dans le chapitre II et les détails de nos travaux sur le sujet font l'objet du chapitre III. Le but de cette section est donc simplement d'introduire la notion de rythmes sensori-moteurs.

Principe de fonctionnement Les rythmes sensori-moteurs sont des oscillations qui ont lieu dans le cortex sensori-moteur I.21 à deux bandes de fréquences :

- la bande de fréquences μ : environ 8-12Hz,
- la bande de fréquences β : environ 18-24Hz.

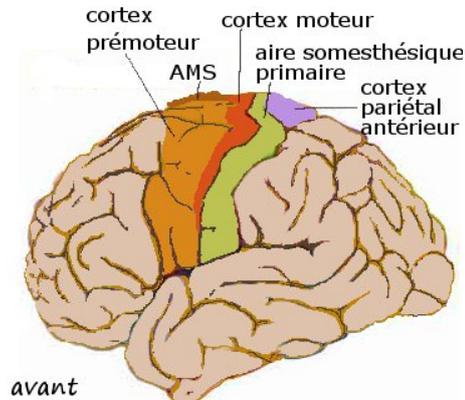


Fig. I.21 – *Le cortex sensori-moteur.* (wikipédia)

Ces oscillations sont modulées en puissance quand le sujet réalise des actions motrices ou même ne fait que les imaginer (voir figure I.22). Ce sont ces modulations qui sont ensuite traduites en commandes pour réaliser une ICM. Un point important est que le fait d'effectuer des mouvements réels ou de simplement les imaginer crée le même type de modulation des rythmes μ et β . En effet, des études par IRMf [Porro *et al.*, 1996] et TEP [Grafton *et al.*, 1996] montrent que les mêmes réseaux neuronaux sont utilisés durant l'exécution de mouvements ou leur imagination.

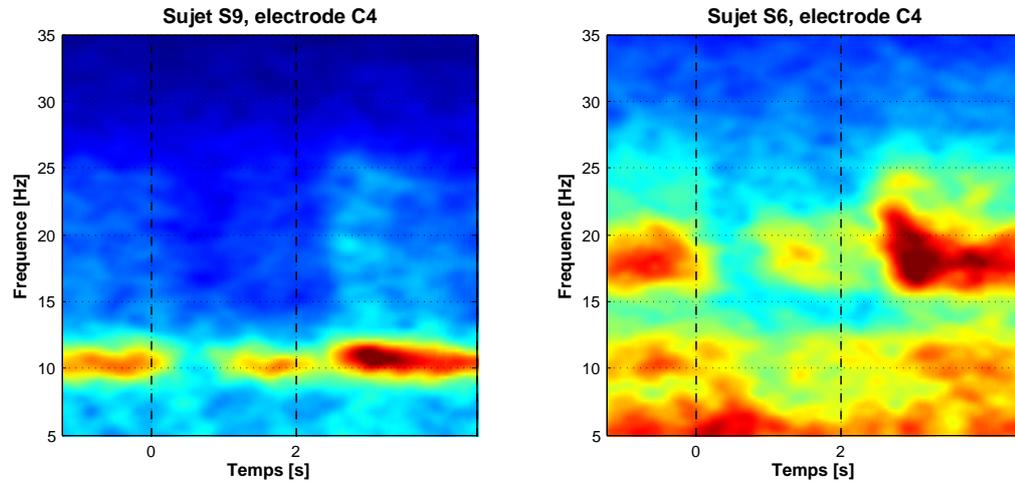


Fig. I.22 – Cartes temps-fréquences moyennes enregistrées au dessus du cortex sensorimoteur droit. Les sujets imaginent des mouvements de la main gauche de $t = 0$ à $t = 2s$. À gauche : la bande de fréquence μ est particulièrement visible et l'on y distingue clairement une diminution de puissance (ERD) durant l'imagination du mouvement suivi d'une augmentation (ERS). À droite : l'ERD sur les deux bandes μ et β est visible, par-contre l'ERS n'est visible que sur le β .

Pour rentrer un peu plus dans les détails, quand aucun mouvement n'est effectué (ou imaginé), les neurones des aires motrices se synchronisent et oscillent. Ceci est dû à un couplage électrique par l'intermédiaire de neurones inhibiteurs impliquant le cortex mais aussi le noyau réticulaire du thalamus [Long *et al.*, 2004, MacKay, 2005]. Ce sont ces oscillations de larges populations de neurones qui sont mesurables en EEG sous la forme des rythmes sensorimoteurs μ et β .

Quand un mouvement est effectué (ou simplement imaginé), les neurones qui participent à l'élaboration de ce mouvement se désynchronisent. Cela se traduit par une diminution de la puissance aux fréquences μ et β mesurée en EEG. Cette diminution de la puissance est appelé **ERD** pour *Event Related Desynchronisation*.

Une fois le mouvement terminé, une augmentation de la puissance dans le μ et β supérieure à la normale peut être mesurée. On l'appelle **ERS** (pour *Event Related Synchronisation*) et l'on parle parfois de *Rebond*. On ne sait pas encore clairement si cette augmentation de la puissance n'est due qu'à la re-synchronisation de la même population de neurones "utilisée" durant le mouvement ou si une autre population entre en jeu.

Il est donc possible en mesurant la puissance aux bandes de fréquences μ et β au dessus des aires motrices de déterminer si le sujet réalise un mouvement (qu'il soit réel ou imaginé).



Fig. I.23 – Flipper commandé par une ICM. Démonstration du Berlin brain-computer interface group au CeBit exhibit à Hannover (Allemagne) [Tangemann et al., 2009]. Pour actionner la commande de droite il suffit au joueur d’imaginer un mouvement de la main droite et pour actionner celle de gauche il doit penser à bouger sa main gauche. Le simple fait que le sujet imagine un mouvement de ses mains fait varier la puissance dans les bandes de fréquences μ et β ce qui est traduit en commandes par l’ICM.

Bases anatomiques : organisation somatotopique du cortex sensori-moteur

La planification, le contrôle et l’exécution des mouvements volontaires sont effectués par le cortex moteur (figure I.21). L’organisation du cortex moteur est loin d’être aléatoire et suit ce qu’on appelle l’organisation somatotopique : à chaque région du cortex moteur correspond une région du corps et les régions voisines du corps ont généralement des représentations proches sur le cortex. Cette organisation a été mise en évidence au début du 19^{ème} siècle chez les grands singes et dans les années 30 chez l’homme par Wilder Penfield [Penfield et Boldrey, 1937].

Il est donc possible de mettre en correspondance avec chaque partie du cortex moteur les ensembles de muscles qui y sont commandés¹⁵ (voir figure I.24 sur l’homonculus de Penfield). Ainsi en déterminant sur quelle région du cortex la puissance dans le μ et le β a varié on peut non seulement essayer de déterminer si l’utilisateur a effectué un mouvement ou non, mais aussi de retrouver avec quelle partie du corps. Ceci permet donc d’augmenter le nombre de signaux de contrôle de l’ICM.

C’est là que l’on comprend tout l’avantage d’une technique d’acquisition ayant une bonne résolution spatiale. Par exemple avec l’IRMf qui a une résolution spa-

¹⁵D’après des travaux plus récents [Schieber et Hibbard, 1993, Barinaga, 1995], même si les cartes motrices restent exactes du point de vue de leur topographie générale, les détails plus fins sont beaucoup plus complexes. En effet, les colonnes verticales du cortex moteur ne contrôlèrent pas les muscles individuellement, mais plutôt des mouvements.

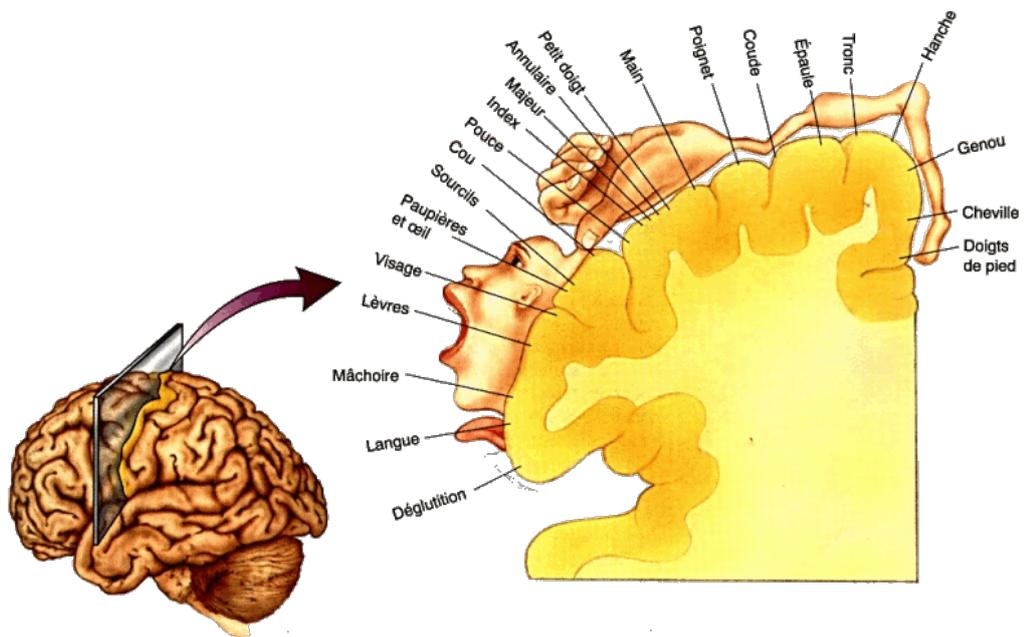


Fig. I.24 – L'homonculus de Penfield [Penfield et Rasmussen, 1950]. On représente pour chaque région du cortex moteur la région du corps correspondante. Une stimulation électrique de l'aire corticale de la main droite provoque des mouvements de la main droite. Inversement quand un sujet bouge ses pieds cela fait varier la puissance dans le μ et le β des aires corticales des pieds.

tiale de l'ordre du millimètre, il est possible de distinguer des mouvements de l'index de ceux du pouce. Avec l'EEG on distingue des mouvements de la main droite de ceux de la main gauche (les aires corticales associées se trouvant chacune sur l'hémisphère controlatéral¹⁶) ou encore des pieds. Les aires corticales des pieds se trouvant repliées entre les deux hémisphères (figure I.24), il n'est pas possible de distinguer des mouvements du pied droit de ceux du pied gauche [Fruitet et Clerc, 2009].

L'un des objectifs de cette thèse est d'essayer d'augmenter la résolution spatiales ICM basée sur l'EEG en résolvant un problème inverse en espérant que cela permette une meilleure discrimination. Les travaux sur le sujet font l'objet du chapitre V de ce manuscrit.

Modulation de l'intensité des commandes Jusqu'à présent on a parlé d'utiliser les rythmes sensori-moteur uniquement pour retrouver quelle action motrice a été réalisée (ou imaginée) par l'utilisateur. Il s'agit donc d'extraire à partir d'une fenêtre de signal une commande parmi un ensemble de commandes possibles. Cependant les modulations des rythmes sensori-moteurs peuvent aussi être converties en signaux de contrôle d'intensité variable.

Prenons par exemple l'ICM développée au Wadsworth Center par Dennis McFarland [Wolpaw et McFarland, 2004]. Il s'agit de contrôler le mouvement d'un curseur sur un écran d'ordinateur en utilisant la modulation des rythmes sensori-moteurs. Le principe de base durant les premières sessions d'apprentissage consiste à imaginer un mouvement (par exemple de la main droite) pour déplacer le curseur dans une première direction (vers la droite) versus un autre mouvement (comme la main gauche) pour le déplacer dans le sens opposé (vers la gauche). Les modulations des rythmes sensori-moteurs sont utilisés pour moduler la vitesse de déplacement du curseur. La conversion des features en signal de contrôle est alors effectuée par une régression linéaire et plus une classification. En suivant le même principe l'utilisateur apprend ensuite à contrôler le mouvement du curseur dans le sens vertical, puis combine l'ensemble des commandes pour contrôler le mouvement du curseur en deux dimensions [Wolpaw et McFarland, 2004]. Cette démarche a été étendue récemment au contrôle d'un curseur en trois dimensions [McFarland *et al.*, 2010].

Avantages et inconvénients des ICM utilisant les rythmes sensori-moteurs Un des principaux inconvénients des ICM utilisant les rythmes sensori-moteurs est qu'elles nécessitent souvent de longues sessions d'apprentissage pour que l'utilisateur arrive à un degré de contrôle suffisant. En contre-partie plus l'utilisateur va se servir de l'ICM plus ses performances vont s'améliorer.

Utiliser les rythmes sensori-moteurs provenant du cortex pour contrôler une

¹⁶la main droite est commandée par l'hémisphère gauche et vice et versa

ICM peut paraître naturel au sens où le cortex moteur est “fait pour”¹⁷ envoyer des commandes vers des représentations internes de mouvements. On remarquera qu’il est capable de s’adapter à de nouvelles interfaces (souris, clavier...) et qu’avec le temps leur utilisation devient “naturelle”. Dans une certaine, mesure il en est de même pour les ICM basées sur les rythmes sensori-moteur. Après un temps d’apprentissage plus ou moins long, l’utilisation de l’interface devient naturelle. Il n’est plus nécessaire pour envoyer une commande de penser à bouger sa main droite mais simplement désirer déplacer le curseur vers la droite (comme c’est le cas quand on utilise une souris) [Wolpaw *et al.*, 2002].

Il faut cependant se rappeler que lors de mouvements réels les cellules pyramidales (d’où proviennent les rythmes sensori-moteur enregistrés en EEG) ne communiquent pas directement avec les muscles mais passent par des intermédiaires et qu’il existe de nombreuses boucles de rétro-action et de régulation afin d’obtenir une commande propre et précise (figure I.3 page 17). Cette absence devra être compensée pour une ICM par un travail important sur la conversion du signal de contrôle (intrinsèquement peu précis et bruité) en commandes [Wolpaw, 2007].

4.2.3 Reconnaissance de tâches cognitives

Tâches cognitives pouvant être utilisées en sus de l’imagination motrice De la même façon que des tâches motrices réalisées avec différentes parties du corps génèrent des types d’activité cérébrale qui sont distinguables grâce à l’EEG, la réalisation de tâches cognitives suffisamment différentes peut être utilisée pour contrôler une ICM.

Certaines tâches cognitives sont couramment utilisées pour réaliser des ICM.

- **Calcul mental** : le sujet réalise des tâches itératives de calcul mental simple, de type $23 + 18 + 18...$ ou $13 \times 4 \times 4...$. Cela met en jeu principalement les aires frontales [Millan et Mourino, 2003, Besserve *et al.*, 2011].
- **Imagination visuo-spatiale** : par exemple imaginer la rotation d’un cube dans l’espace [Millan et Mourino, 2003, Besserve *et al.*, 2011].
- **Imagination auditive** : il s’agit principalement d’imagination musicale [Besserve *et al.*, 2011].
- **Linguistique** : par exemple trouver des mots commençant par une même lettre, ou faire des associations de mots [Millan et Mourino, 2003].

Avantages et inconvénients de la reconnaissance de tâches cognitives Utiliser des tâches cognitives variées sans se limiter aux tâches motrices a deux avantages :

- Cela permet d’augmenter le nombre d’actions, et donc de commandes, que l’utilisateur peut transmettre à la machine.

¹⁷Malgré ce que sous-entend l’emploi de “fait pour” je tiens à préciser que je ne crois pas aux théories créationnistes, mais j’en profite pour vous recommander chaudement de lire “Les dinosaures et la bible” (premier lien dans google)

- Les aires cérébrales et les processus neurologiques mis en jeu lors des diverses tâches étant très différents, les taux de discrimination entre les différentes commandes sont généralement supérieurs à ceux entre tâches motrices.

Cependant, utiliser des tâches cognitives autres que l'imagination motrice implique certaines contraintes et comporte son lot d'inconvénients :

- **Plus de modulation de l'intensité des commandes** Dans la section dédiée à l'imagination motrice on a expliqué que les tâches d'imagination motrice pouvaient être modulées en intensité. Ce n'est pas le cas pour la plupart des tâches cognitives. On peut donc accroître le nombre de commandes, mais on reste limité à un système de type détection de tâches et on se prive d'une modulation en intensité des différentes commandes.
- **ICM moins naturelle** Utiliser des tâches cognitives variées pour contrôler une ICM est, *a priori*, moins naturel qu'utiliser l'imagination motrice. Comme on l'a expliqué précédemment, après une longue période d'apprentissage, l'utilisation d'une ICM basée sur l'imagination motrice devient naturelle. Le sujet n'a plus besoin d'imaginer des mouvements pour contrôler l'ICM et envoie "naturellement" des commandes. Il n'est pas sûr qu'une ICM requérant du sujet qu'il s'imagine une danseuse réalisant des entrechats puisse devenir naturelle.

État de l'art de quelques problématiques liées aux ICM

CE chapitre aborde de façon plus approfondie certaines problématiques importantes liées aux ICM sur lesquelles nous avons travaillé. Après une brève introduction, nous détaillerons les différentes utilisations de l'imagination motrice dans les ICM, puis nous effectuerons un état de l'art des méthodes qui peuvent être utilisées pour augmenter la résolution spatiale de l'EEG.

Sommaire

1	Introduction	49
1.1	Nouvelles ICM et interfaces de contrôle	49
1.2	Extraction du signal de contrôle	49
2	Différentes façons d'utiliser l'imagination motrice	50
2.1	Contrôle multidimensionnel modulé en amplitude	51
2.2	Utilisation synchrone de l'imagination motrice	52
2.3	Détection asynchrone de tâches	52
2.3.1	<i>Détection continue de tâches d'imagination motrice</i>	53
2.3.2	<i>Interrupteur commandé par la pensée</i>	54
2.3.3	<i>Bouton(s) commandé(s) par la pensée</i>	55
2.4	Sélection de tâches adaptées au contrôle d'une ICM	56
2.4.1	<i>Le problème de la sélection de tâches</i>	56
2.4.2	<i>État d'un art encore inexistant</i>	56
2.4.3	<i>Vers une méthode automatique de la sélection de tâche</i>	57
3	Augmentation de la résolution spatiale de l'EEG	58
3.1	Filtrage spatial laplacien	58
3.2	Le <i>Common Spatial Filter</i> ou CSP	59
3.3	Reconstruction de sources	61
3.3.1	<i>Reconstruction de sources aveugle</i>	61
3.3.2	<i>Reconstruction de sources avec modélisation anatomique</i>	62

1 INTRODUCTION

Il a été expliqué dans le chapitre précédant que les ICM permettent de créer une nouvelle voie de communication. Malheureusement, à l'heure actuelle ce canal de communication est tellement limité et peu fiable, que cela réduit considérablement l'utilité réelle des ICM. Même dans le cas de patients atteints de graves paralysies ou traumatismes, les ICM ne sont utiles qu'en dernier recours, quand aucun moyen de communication naturel n'est plus possible.

Un des enjeux majeurs pour le développement des ICM est donc de pouvoir augmenter leur capacité de transfert d'information et leur robustesse. Ce gain de performances doit être obtenu en améliorant chaque partie de la chaîne constituant une ICM (voir schéma I.1). On peut cependant dégager deux axes de développement particulièrement représentés dans la recherche sur ce sujet :

- la création de nouvelles ICM et le travail sur l'interface de contrôle,
- l'amélioration du transducteur et en particulier de l'extraction du signal de contrôle (voir schéma I.2).

1.1 Nouvelles ICM et interfaces de contrôle

La multiplication des laboratoires de recherche travaillant sur les ICM a conduit à une multiplication des paradigmes. En effet, bien que l'éventail des phénomènes physiologiques permettant de réaliser des ICM soit restreint, les façons de les utiliser sont en constante augmentation. Des mêmes phénomènes, comme la production de l'ERD/ERS par l'imagination de tâches motrices, peuvent être utilisés dans des protocoles divers donnant lieu à des signaux de contrôles totalement différents (voir section 2). D'autres, comme le P300, sont utilisés dans des protocoles très similaires, et emploient donc les mêmes transducteurs, mais pour aboutir à des signaux de contrôle totalement différents, allant du clavier virtuel [Farwell et Donchin, 1988] au logiciel de peinture [Munssinger *et al.*, 2010] en passant par la commande de fauteuil roulant [Galán *et al.*, 2008].

Il est à noter l'apparition d'ICM hybrides basées soit sur différentes ICM, soit couplant une ICM à une interface traditionnelle. L'intégration de différentes ICM permet ainsi de combiner la souplesse d'ICM asynchrones aux plus grandes performances fournies par les ICM synchrones comme celles basées sur le SSVEP [Pfurtscheller *et al.*, 2010b]. Un état de l'art sur les différentes ICM hybrides est effectué dans [Pfurtscheller *et al.*, 2010a].

1.2 Extraction du signal de contrôle

Pré-traitements Le signal de contrôle peut être amélioré par l'utilisation de pré-traitements. Il peut s'agir de détecter et de supprimer des artefacts afin de rendre l'interface plus fiable ou d'augmenter le rapport signal sur bruit.

Différentes techniques permettant d'augmenter la résolution spatiale de l'EEG sont présentées dans la section 3 page 58. Le chapitre V est consacré au travail que nous avons effectué sur ce sujet.

Extraction et conversion des features De nombreux travaux ont été effectués pour améliorer chaque étape de l'extraction du signal de contrôle et en particulier pour la conversion des features. La classification et la régression (utilisés pour la conversion des features en signal de contrôle) sont des problèmes d'apprentissage supervisé pour lesquels il existe de nombreux algorithmes. Ainsi, un grand nombre de méthodes provenant du domaine de l'apprentissage automatique ont été appliquées aux ICM permettant d'augmenter légèrement mais régulièrement les taux de détection et de classification. L'organisation de *BCI competitions*, avec la mise à disposition de jeux de données, permet des comparaisons des différentes méthodes et algorithmes développés.

Pendant le cas des ICM est un cas très particulier de l'apprentissage supervisé, nécessitant à mon sens le développement d'algorithmes spécifiquement conçus pour répondre à plusieurs contraintes :

- **Des données très bruitées** : l'information que l'on cherche à retrouver dans les features est généralement simple mais noyée dans le bruit.
- **Non stationnarité** : le signal mesuré, et donc les features qui en sont extraites, évoluent au cours d'une même session d'enregistrement et peuvent varier grandement entre les sessions.
- **Un ensemble de données très réduit** : les données devant être enregistrées sur des sujets humains leur nombre est forcément très limité. Ce nombre de données limité empêche l'utilisation de modèles plus complexes.

Bien que ce soit un domaine important, en particulier pour prendre en compte la non-stationnarité des features, nous n'avons pas porté d'attention particulière sur la conversion des features durant cette thèse.

2 DIFFÉRENTES FAÇONS D'UTILISER L'IMAGINATION MOTRICE

De nombreuses ICM utilisent l'imagination de tâches motrices pour générer le signal de contrôle. Bien que ces ICM soient généralement basées sur les mêmes phénomènes physiologiques, l'utilisation de ces dernières peuvent être très différentes. Il est en effet possible d'utiliser l'imagination motrice pour réaliser des ICM synchrones ou asynchrones et obtenir un signal de contrôle discret ou continu.

- Utilisation **asynchrone** : l'utilisateur envoie les commandes quand il le désire.

- Utilisation **synchrone** : l'utilisateur ne peut contrôler l'ICM qu'aux instants où sa réponse est attendue.
- Signal de contrôle **discret** : le nombre de commandes que l'utilisateur peut envoyer est fini. Il s'agit soit de la détection d'une tâche (signal de contrôle binaire) soit de la discrimination entre plusieurs tâches.
- Signal de contrôle **continu** : le sujet peut moduler l'amplitude de sa commande, qui peut être d'une ou plusieurs dimensions.

2.1 Contrôle multidimensionnel modulé en amplitude

L'imagination motrice a plusieurs avantages qui la rendent particulièrement adaptée au contrôle continu d'une ICM :

- différents mouvements peuvent être imaginés afin de transmettre un signal de contrôle à plusieurs dimensions,
- les commandes envoyées peuvent être modulées en intensité,
- les commandes peuvent être utilisées de façon continue et asynchrone,
- si l'ICM est bien conçue, son utilisation peut être "naturelle" pour le sujet¹.

L'équipe de J. Wolpaw développe une ICM basée sur l'imagination motrice qui tire parti des ces avantages. Elle permet à l'utilisateur de contrôler un curseur sur un écran d'ordinateur en 1, 2 et maintenant 3 dimensions grâce à l'EEG [Wolpaw *et al.*, 1991, Wolpaw et McFarland, 2004, McFarland *et al.*, 2010] et 2 dimensions avec de l'ECOG [Schalk *et al.*, 2008].

De façon surprenante, le traitement de cette ICM utilisée pour transformer le signal biologique en signal de contrôle est relativement simple et n'implique pas d'algorithmes d'apprentissage très sophistiqués [Wolpaw et McFarland, 2004]. La plus grande partie de l'apprentissage permettant le contrôle de cette ICM est en fait réalisée par le sujet lui-même.

Le mouvement du curseur dans chaque dimension est contrôlé par une équation linéaire faisant intervenir les puissances dans les bandes de fréquences μ et/ou β mesurées au dessus des hémisphères droit et gauche du cortex sensorimoteur. Par exemple, le mouvement vertical du curseur est déterminé par l'équation :

$$M_V = a_V(w_{RV}R_V + w_{LV}L_V + b_V)$$

où R_V (L_V) est la puissance mesurée au dessus de l'hémisphère droit (et gauche respectivement) dans la bande μ ou β . Afin d'améliorer les performances et de simplifier l'apprentissage de l'utilisateur les paramètres a_V et b_V sont adaptés en ligne [McFarland *et al.*, 1997a]. La puissance est mesurée à l'aide d'une analyse

¹au sens où imaginer un mouvement de la main droite parait adapté pour déplacer un curseur vers la droite

fréquentielle autorégressive [McFarland et Wolpaw, 2008] après l'application d'un filtre laplacien [McFarland et al., 1997b].

Le sujet apprend à contrôler chaque direction indépendamment, avant d'apprendre le contrôle du mouvement en deux voire trois dimensions. D'après [Wolpaw et al., 2002], après 2-3 semaines d'apprentissage à raison de 2 à 3 sessions de 40 minutes par semaine, environ 80% des sujets arrivent à un degré de contrôle significatif. L'utilisation de l'ECOG permet d'obtenir un apprentissage beaucoup plus rapide (30 minutes pour un contrôle sommaire en 2 dimensions) [Schalk et al., 2008].

2.2 Utilisation synchrone de l'imagination motrice

Contrairement à la modulation continue des rythmes moteurs, qui implique un apprentissage particulièrement long de la part du sujet, l'utilisation synchrone de l'imagination motrice simplifie grandement l'expérimentation. La plupart des ICM basées sur l'imagination motrice utilise donc un protocole synchrone.

L'exemple le plus connu est l'ICM développée à Graz sous le nom de *Graz BCI* [Pfurtscheller et al., 2000, Pfurtscheller et al., 2003]. Cette ICM est basée sur la détection de l'ERD² et utilise un protocole synchrone faisant apparaître des consignes (ou *cues*). Le sujet imagine différents mouvements à des intervalles de temps précis durant plusieurs secondes. La détection du type de mouvement permet d'envoyer la commande. Contrairement à l'ICM développée au Wadsworth Center, une chaîne de traitement plus complexe est utilisée pour classifier les différents mouvements et le sujet a moins besoin d'apprendre à moduler ses rythmes moteurs. Il en résulte que la prise en main est plus rapide : après 6-7 sessions, les utilisateurs peuvent atteindre des performances dépassant 90% de réussite pour une ICM à deux commandes (i.e. imagination de la main gauche vs. main droite) et environ 90% des utilisateurs peuvent utiliser ce système [Wolpaw et al., 2002]. Cependant, l'utilisation d'un protocole synchrone est particulièrement limitante.

2.3 Détection asynchrone de tâches

Jusqu'à récemment, relativement peu d'intérêt avait été porté aux ICM totalement asynchrones [Mason et Birch, 2000]. Cela peut s'expliquer par le fait qu'il est beaucoup plus difficile de concevoir, d'ajuster et d'évaluer des systèmes asynchrones que synchrones. Pourtant du point de vue de leur utilisation, des systèmes asynchrones sont moins contraignants et parfois même indispensables.

Le nombre d'ICM asynchrones basées uniquement sur l'imagination motrice étant réduit, nous avons inclus dans cet état de l'art des ICM utilisant d'autres tâches cognitives.

²Event Related Desynchronisation : baisse de puissance dans les bandes de fréquences μ et β durant le mouvement ou l'imagination d'un mouvement.

Nous avons identifié trois types d'ICM asynchrones. Tout d'abord des ICM détectant le passage d'une tâche mentale à une autre, que nous appellerons **détection continue de tâches mentales**. Deux autres types d'ICM détectent la réalisation de tâches spécifiques au milieu de périodes de repos permettant ainsi au sujet d'envoyer une commande quand il le désire. Bien que parfois regroupés sous le nom de "brain switch", ces deux derniers types d'ICM asynchrones ont des objectifs très différents. Nous avons préféré redéfinir la terminologie habituellement employée afin de distinguer :

- un **interrupteur** commandé par la pensée, ou *brain-controlled switch*, qui permet à un utilisateur d'activer ou de désactiver l'utilisation de l'ICM. Le critère est alors la **robustesse**.
- un **bouton** commandé par la pensée, ou *brain-controlled button*, qui sert à envoyer une commande quand l'utilisateur le désire. Une telle ICM doit être **réactive**.

Le terme de *brain-controlled switch* fut introduit par Mason pour désigner ce que nous appelons un "bouton" mais fut ensuite réutilisé par Pfurtscheller dans le sens d'interrupteur³.

Passons maintenant en revue chacun de ces trois types d'ICM asynchrone.

2.3.1 Détection continue de tâches d'imagination motrice

Il s'agit de l'extension naturelle de l'utilisation synchrone de l'imagination motrice à un protocole asynchrone. Au lieu de distinguer des essais durant lesquels le sujet réalise différentes tâches motrices, il faut classifier l'activité EEG en continu. Dans [Scherer *et al.*, 2008] les sujets arrivent à se déplacer dans un environnement virtuel à l'aide de trois tâches d'imagination motrice (permettant de tourner à droite, à gauche et d'avancer) après avoir effectué un apprentissage grâce à un protocole synchrone. Il est à noter que l'évaluation de telles ICM asynchrones où le sujet est totalement libre est particulièrement difficile.

Des ICM utilisant l'imagination motrice et d'autres tâches mentales ont été utilisées de façon asynchrone. L'ICM décrite dans [Galán *et al.*, 2008] permet ainsi de contrôler la trajectoire d'une chaise roulante à l'aide de trois tâches (imagination de mouvement de la main gauche, repos et recherche de mots commençant par une même lettre). L'ICM *Adaptive Brain Interface* [Millan *et Mourino*, 2003] permet de reconnaître en continu plusieurs tâches mentales (imagination motrice, calcul, langage...) Quand le système de détection n'est pas sûr de la tâche mentale reconnue, aucune commande n'est envoyée. Ce principe est proche du fonction-

³L'article [Mason *et Birch*, 2000] différencie clairement activation et désactivation du système de l'utilisation d'un bouton nommé alors "asynchronous switch". Pourtant Pfurtscheller reprend le terme de "brain-switch" pour l'ICM dont il se sert pour enclencher l'utilisation d'une ICM basée sur le SSVEP [Pfurtscheller *et al.*, 2010b]. La terminologie utilisée par Pfurtscheller semblant plus répandue, nous avons adopté une définition compatible avec cette dernière.

nement de boutons commandés par la pensée : le sujet peut n’envoyer aucune commande et, quand il le désire, faire son choix parmi un ensemble de commandes.

2.3.2 Interrupteur commandé par la pensée ou “*brain-controlled switch*”

Une des problématiques importantes pour permettre à des personnes paralysées d’utiliser une ICM est qu’elles puissent démarrer et arrêter le fonctionnement de l’interface quand elles le souhaitent. Pour cela une autre ICM est utilisée que l’on nomme *interrupteur* ou *brain-switch*. Une telle interface doit permettre à l’utilisateur d’initier ou d’interrompre la communication quand il le désire mais ne doit pas se déclencher quand il effectue d’autres tâches. Les contraintes sont donc différentes que pour les ICM habituelles, puisque la rapidité et la quantité d’information transmise comptent peu et seule importe la robustesse. Il est en effet primordial que le taux de faux positifs (c’est à dire le déclenchement inopportun de l’interrupteur) reste extrêmement faible, même si cela implique un taux de faux négatifs (échec d’activation de l’interrupteur) relativement élevé.

Divers processus physiologiques ont été utilisés comme base pour développer des *interrupteurs* :

- la modification du rythme cardiaque [Scherer *et al.*, 2007],
- la variation des ondes corticales lentes [Kaiser *et al.*, 2001],
- les SSVEP⁴ [Middendorf *et al.*, 2000],
- l’ERS dans la bande β généré par le mouvement des pieds [Pfurtscheller et Solis-Escalante, 2009, Pfurtscheller *et al.*, 2010b],
- l’imagination motrice détectée par NIRS⁵ [Coyle *et al.*, 2007] .

D’autre part, afin d’augmenter la robustesse de l’interrupteur et en particulier de minimiser l’occurrence de faux positifs, des protocoles particuliers peuvent être développés :

- Dans [Kaiser *et al.*, 2001], les patients doivent enchaîner une séquence de modulations des ondes corticales lentes pour activer l’interrupteur.
- Dans [Qian *et al.*, 2010], l’ERD/ERS produit par l’imagination motrice est moyenné sur plusieurs essais pour atteindre un taux de faux positifs inférieur à 1%.
- Dans [Barachant *et al.*, 2011], l’utilisation de l’imagination motrice permet de faire varier une jauge qui doit dépasser un seuil pendant un certain temps pour activer l’interrupteur.

⁴L’utilisation des SSVEP dans les ICM est expliqué dans la section 4.1.2 du chapitre I.

⁵NIRS : *near infrared spectroscopy* ou imagerie proche infra-rouge, voir la section 3.2.4 du chapitre I.

2.3.3 Bouton(s) commandé(s) par la pensée ou “brain-controlled button(s)”

Mis à part l'intérêt relativement récent sur les *brain-switch*, très peu d'attention est portée sur les ICM totalement asynchrones. Ce type d'ICM permet une utilisation plus naturelle et impose moins de contraintes à l'utilisateur. Mais cette liberté de l'utilisateur rend l'étude et le développement de ces ICM plus complexe. À notre connaissance seule les équipes de Mason et Birch [Mason et Birch, 2000, Bozorgzadeh *et al.*, 2000, Fatourechi *et al.*, 2008]⁶ et de Lécuyer et Lotte [Lotte *et al.*, 2008b] ont réalisé un bouton commandé par la pensée. Cependant, dans [Zhang *et al.*, 2007], les auteurs proposent un algorithme pour détecter les périodes de repos de l'utilisateur entre les tâches d'imaginations motrice, ce qui *in fine* permet de réaliser des boutons commandés par la pensée.

Idéalement de telles ICM permettent à l'utilisateur d'envoyer, quand il le désire, un signal faisant partie d'un ensemble discret de commandes (les boutons). Ce sont donc des ICM à part entière qui fournissent à elles seules un moyen de communication. La section 2 du chapitre III donne un exemple de navigation spatiale basée sur l'utilisation d'un simple bouton commandé par la pensée.

Contrairement aux interrupteurs commandés par la pensée, l'utilisation des boutons doit être suffisamment réactive pour que l'ICM soit utile. D'autre part, les contraintes sur le nombre de faux négatifs et de faux positifs dépendent des utilisations. Il peut en effet être préférable d'accepter quelques déclenchements intempestifs, plutôt que de manquer une activation au moment opportun.

Le low-frequency asynchronous switch design ou LF-ASD Il s'agit d'utiliser l'information contenue dans la bande fréquence 1-4Hz pour détecter des mouvements particuliers de l'index. Durant les premières expériences, l'utilisation d'un protocole totalement asynchrone oblige les expérimentateurs à utiliser des mouvements réels pour savoir quand les sujets réalisent des mouvements [Mason et Birch, 2000]. Par la suite l'imagination de mouvements fut utilisée dans des expériences en-ligne, mais les sujets devaient rapporter les erreurs de détections en appuyant sur un bouton [Bozorgzadeh *et al.*, 2000].

Simulation d'expériences asynchrones Un nombre croissant d'études simulent le fonctionnement d'hypothétiques ICM asynchrones sur des données enregistrées avec des protocoles synchrones. Ceci permet de résoudre un des problèmes majeurs des ICM asynchrones : l'expérimentateur ne sait pas quand le sujet décide d'envoyer sa commande. Cependant le protocole peut être particulièrement éloigné d'un fonctionnement asynchrone, voire ne pas avoir été prévu initialement pour simuler des expériences asynchrones [Townsend *et al.*, 2004, Sadeghian et Moradi, 2007]. Il est alors possible que d'importants biais soient présents (voir par exemple figure III.13 page 95).

⁶Certaines de ces études sont réalisées sur des mouvements réels de l'index, mais l'objectif est de pouvoir commander l'interface par la pensée.

Du rebond bêta aux boutons commandés par la pensée L'équipe de Pfurtscheller utilise l'ERS postérieure à l'imagination de mouvements des pieds (rebond β) pour réaliser un interrupteur commandé par la pensée. Pour utiliser l'ERS produit par l'imagination d'un mouvement, il faut attendre que ce mouvement se termine, ce qui introduit un délai. Cependant, quand des mouvements brefs sont utilisés, ce délai est réduit et n'est pas forcément incompatible avec une utilisation de type bouton [Fruitet *et al.*, 2011, Thomas *et al.*, 2012].

Le chapitre III de ce manuscrit s'intéresse à l'utilisation de l'ERD et de l'ERS pour la réalisation d'un ou plusieurs boutons actionnés par l'imagination de différentes tâches motrices. Il constitue une étude pionnière vers la réalisation d'une ICM asynchrone à multiples commandes discrètes.

2.4 Sélection de tâches adaptées au contrôle d'une ICM

2.4.1 *Le problème de la sélection de tâches*

Les ICM utilisant l'imagination motrice ou la reconnaissance de tâches cognitives sont basées sur la distinction de l'activité cérébrale générée par l'imagination de ces différentes tâches. Ces tâches font intervenir différents réseaux neuronaux, situés dans différentes régions du cortex, et aboutissent à une modification de l'activité cérébrale qu'il est plus ou moins facile de détecter et de différencier. Afin de permettre de bonnes performances, il est important d'utiliser des tâches générant des patterns suffisamment clairs et distincts. Malheureusement les tâches permettant les meilleures performances ne sont pas identiques pour tous les sujets [Le Pavec et Clerc, 2008, Dobrea et Dobrea, 2009, Dobrea *et al.*, 2010]. Il serait donc préférable d'adapter pour chaque sujet les tâches utilisées.

2.4.2 *État d'un art encore inexistant*

C'est avec surprise, après avoir envisagé la sélection de tâches pour les ICM comme un problème à part entière, que nous nous sommes aperçus que quasiment aucune étude ne portait à proprement parler sur le sujet [Dobrea et Dobrea, 2009]. Les équipes travaillant sur les ICM utilisant l'imagination motrice sélectionnent pourtant les tâches qui sont utilisées pour effectuer le contrôle, mais souvent de façon très sommaire.

On peut rencontrer trois types de stratégies :

1. **Choix des tâches basé sur un *a priori*** : Il s'agit simplement d'effectuer un choix basé sur l'expérience, ou sur une recherche bibliographique, des tâches les plus appropriées à l'ICM que l'on veut développer [Pfurtscheller *et al.*, 2005]. L'inconvénient est que les tâches ne sont pas optimisées pour les sujets. Les résultats présentés dans le chapitre III

montrent qu'une tâche donnant de bons résultats pour un sujet peut ne pas convenir pour un autre.

2. **Sélection par élimination** : C'est la méthode la plus rigoureuse. Il s'agit de réaliser la première expérience avec un nombre de tâches légèrement supérieur à celui nécessaire pour contrôler l'ICM, puis d'éliminer les tâches donnant les moins bons résultats [Scherer *et al.*, 2008, Dobrea et Dobrea, 2009, Thomas *et al.*, 2012].

L'inconvénient est qu'on ne peut pas essayer beaucoup de tâches différentes pour ne pas trop allonger la durée de l'expérience. De plus, une grande partie des données acquises devient inutile puisqu'elle ne sert qu'à éliminer les tâches qui ne seront plus utilisées.

Exemple

*Il est fréquent, pour une ICM nécessitant deux tâches d'imagination motrice, de faire la première expérience avec la main droite, la gauche, les pieds et la langue et d'éliminer la tâche la moins performante [Scherer *et al.*, 2008].*

3. **Sélection par pré-étude empirique** : Certaines équipes ont pour habitude de faire une pré-expérience durant laquelle un grand nombre de tâches sont essayées. Les meilleures tâches sont alors utilisées durant les expériences suivantes. C'est le cas, par exemple, dans l'équipe de Jonathan Wolpaw dans laquelle j'ai effectué un stage avant ma thèse. Il est à noter que le protocole utilisé pour cette première expérience d'exploration était assez vague et qu'il n'est pas sûr que le nombre de répétitions de chaque tâche permette vraiment de déterminer lesquelles sont les plus appropriées.

2.4.3 Vers une méthode automatique de la sélection de tâche

Le choix des tâches d'imagination motrice, ou d'une façon plus générale de tâches cognitives, est un aspect important de la réalisation d'une ICM. A l'heure actuelle, aucune méthode n'existe pour sélectionner de façon efficace et fiable les tâches les plus adaptées à chaque sujet. Il serait pourtant intéressant de pouvoir, pour chaque sujet, évaluer une vaste variété de tâches pour sélectionner celles qui permettent les meilleures performances.

Le chapitre IV de ce manuscrit est dédié au travail que nous avons effectué sur le développement d'une méthode permettant une sélection automatique de tâches. Les objectifs sont de permettre une exploration d'un grand nombre de tâches en peu de temps tout en garantissant une évaluation suffisamment fiable pour déterminer les meilleures tâches.

3 AUGMENTATION DE LA RÉOLUTION SPATIALE DE L'EEG

Une des problématiques majeures des ICM utilisant l'EEG est la faiblesse du taux de transfert d'information⁷. Cette limitation est en grand partie due au fait que le signal enregistré par EEG contient peu d'informations utiles et est extrêmement bruité. En attendant une révolution majeure au niveau de la technique d'enregistrement, plusieurs méthodes ont été développées pour augmenter la résolution spatiale de l'EEG, c'est à dire pour permettre de distinguer des phénomènes indiscernables dans le signal brut.

3.1 Filtrage spatial laplacien

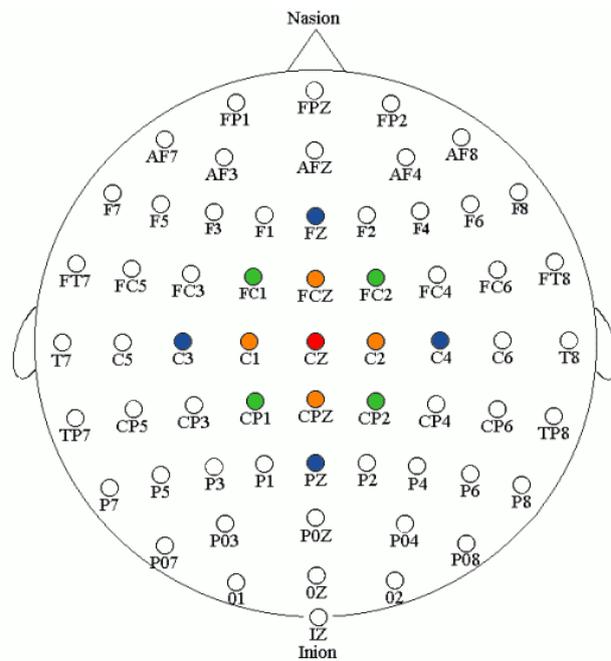


Fig. II.1 – Nom et positionnement des 64 électrodes suivant le système 10-20. En couleur sont représentés trois filtres laplaciens spatiaux appliqués à l'électrode Cz (rouge). La moyenne du signal mesuré en C3, C4, Fz et Pz (bleu) est retirée à l'électrode Cz (rouge) pour le filtre laplacien large. Les électrodes utilisées pour un filtre laplacien moyen et petit sont représentées en vert et orange respectivement.

C'est la méthode la plus simple et la plus utilisée. Il s'agit de soustraire au signal mesuré à chaque électrode la moyenne des signaux mesurés sur les électrodes voisines (voir figure II.1). L'utilisation d'un filtre laplacien permet d'augmenter les performances d'une ICM [McFarland *et al.*, 1997b]. Cela peut être expliqué par deux raisons.

⁷Le taux de transfert d'information est défini à la page 80.

1. Une plus grande précision sur l'activité corticale. Comme expliqué précédemment, les signaux mesurés sur une électrode proviennent d'une région spatialement étendue du cortex et subissent un filtrage lié au crâne. L'utilisation d'un filtre laplacien peut permettre de se focaliser plus précisément sur l'activité qui se trouve sous chaque électrode.
2. Un filtrage du bruit. Le bruit présent de façon uniforme sur les électrodes proches est éliminé par l'utilisation d'un filtre laplacien. L'utilisation de la référence moyenne⁸ permet aussi de supprimer le bruit uniforme sur l'ensemble des électrodes (en particulier les bruits extérieurs). Cependant un bruit non uniforme peut être amplifié par la référence moyenne sur certaines électrodes alors qu'un filtrage laplacien le supprimerait. Par exemple les clignements des yeux provoquent d'importants artefacts sur les électrodes frontales qui peuvent être répercutés sur les autres électrodes par la référence moyenne.

3.2 Le Common Spatial Filter ou CSP

Le CSP est une méthode permettant de construire des filtres spatiaux optimisant la séparation de deux classes. Elle a été développée par [Fukunuga, 1990], puis utilisée pour l'extraction de composantes anormales dans les signaux EEG [Koles, 1991] avant d'être introduite par [Ramoser *et al.*, 2000] dans le traitement des ICM où elle est aujourd'hui massivement utilisée. Durant la *BCI competition* de 2005, tous les vainqueurs excepté un ont utilisé le CSP ou un algorithme dérivé [Blankertz *et al.*, 2006].

Filtres spatiaux Notons $\mathbf{m}(t) = [m_1(t) \dots m_{N_E}(t)]^T$ le vecteur des mesures effectuées à l'instant t sur l'ensemble des N_E électrodes. Un filtre spatial $\mathbf{w} = [w_1 \dots w_{N_E}]^T$ est défini par :

$$s_{\mathbf{w}}(t) = \sum_{e=1}^{N_E} w_e m_e(t) = \mathbf{w}^T \mathbf{m}(t)$$

Optimisation des filtres par le CSP On commence généralement par effectuer un filtrage fréquentiel sur la bande contenant l'information (environ 8-30Hz pour des données d'imagination motrice). On calcule ensuite la matrice de covariance $\Sigma_i \in \mathbb{R}^{N_E \times N_E}$ des signaux mesurés sur chaque canal pour chaque essai i . On définit une matrice de covariance moyenne pour chaque classe $k \in \{1, 2\}$ par :

$$\bar{\Sigma}_k = \frac{1}{\text{Card}(\mathcal{I}_k)} \sum_{i \in \mathcal{I}_k} \Sigma_i$$

avec \mathcal{I}_k les indices des essais de la classe k .

⁸La moyenne du signal mesuré sur l'ensemble des électrodes est retirée au signal de chaque électrode.

Le but du CSP est de construire un ensemble de filtres spatiaux qui maximise l'écart entre les moyennes des covariances. Pour cela, on effectue la diagonalisation conjointe des matrices $\bar{\Sigma}_1$ et $\bar{\Sigma}_2$ avec une contrainte permettant l'unicité :

$$\begin{cases} \mathbf{W}^T \bar{\Sigma}_1 \mathbf{W} = \mathbf{D}_1 \\ \mathbf{W}^T \bar{\Sigma}_2 \mathbf{W} = \mathbf{D}_2 \\ \text{avec } \mathbf{D}_1 + \mathbf{D}_2 = \mathbf{I} \end{cases}$$

Supposons que les valeurs propres de \mathbf{D}_1 soient ordonnées par valeurs décroissantes, on a alors :

$$\begin{aligned} \mathbf{D}_1 &= \text{diag}(\lambda_i)_{i \in \{1..N_E\}} \\ \mathbf{D}_2 &= \text{diag}(1 - \lambda_i)_{i \in \{1..N_E\}} \\ &\text{et } \lambda_1 \geq \lambda_i \geq \lambda_{N_E} \end{aligned}$$

La matrice $\mathbf{W} = [\mathbf{w}_1.. \mathbf{w}_{N_E}]$ est composé de N_E filtres spatiaux. Après application du i ème filtre spatial \mathbf{w}_i , la moyenne de la variance sur les différents essais est λ_i sur la classe 1 et $1 - \lambda_i$ sur la classe 2.

Les premiers et derniers filtres, associés aux plus grandes et plus petites valeurs propres, maximisent ainsi l'écart des moyennes des variances du signal filtré sur les deux classes.

Avantages et inconvénients du CSP L'avantage du CSP est qu'il permet, en se basant sur des données d'apprentissage, de déterminer des filtres spatiaux spécifiquement conçus pour faciliter la discrimination entre deux classes. Son utilisation a permis d'améliorer les taux de classification sur certaines expériences et il est maintenant largement utilisé dans les ICM. Ce procédé comporte cependant plusieurs inconvénients :

- Il utilise pour être efficace un grand nombre d'électrodes. Cela rend l'ICM moins ergonomique et rallonge la mise en place du dispositif.
- Si une des électrodes est bruitée, ou ne fonctionne plus correctement, cela contamine tout le dispositif.
- L'optimisation des filtres n'est pas robuste⁹.
- L'utilisation de l'information des classes rend la méthode sujette à un risque de sur-apprentissage.

Pour pallier certains de ces problèmes, des généralisations du CSP ont été développées, en particulier au cas multi-classes [Grosse-Wentrup et Buss, 2008] [Yan et al., 2008] et à l'extraction de filtres spatiaux-temporels [Dornhege et al., 2006, Ang et al., 2009]. Différentes améliorations permettent aussi de rendre le CSP plus robuste

⁹C'est à dire qu'une petite perturbation du signal peut modifier grandement les filtres.

[Wu *et al.*, 2009, Ang *et al.*, 2009]. Barachant propose une amélioration du CSP en effectuant les calculs dans l'espace des matrices symétriques définies positives [Barachant *et al.*, 2011].

3.3 Reconstruction de sources

Le signal enregistré en EEG provient de l'activité de populations de neurones du cortex (voir la section 2 du premier chapitre page 19). Ces populations émettent des champs électromagnétiques qui se propagent à travers les différents tissus avant d'être intégrés sur les capteurs pour constituer le signal mesuré. On peut modéliser l'activité de ces populations par des dipôles de courant, appelés *sources*, puis tenter de reconstruire à partir du signal mesuré l'activité de chaque dipôle ou source.

Une méthode permettant une bonne reconstruction de l'activité des sources permet de distinguer des phénomènes ayant des origines proches sur le cortex, ce qui serait extrêmement utile pour augmenter la précision des ICM.

3.3.1 Reconstruction de sources aveugle

Une première modélisation du problème consiste à supposer que le signal mesuré provient d'un mélange linéaire inconnu d'un ensemble de sources fixées :

$$\mathbf{M}(t) = \mathbf{A}\mathbf{S}(t)$$

avec $\mathbf{A} \in \mathbb{R}^{N_E \times N_S}$ la matrice de mélange et $\mathbf{S}(t) = [s_1(t) \dots s_{N_S}(t)]^T$ l'activité des N_S sources à l'origine des mesures $\mathbf{m}(t) = [m_1(t) \dots m_{N_E}(t)]^T$.

Il s'agit alors de déterminer une matrice de *démixage* $\mathbf{W} \in \mathbb{R}^{N_S \times N_E}$ permettant de construire une estimation $\hat{\mathbf{S}}(t)$ de l'activité des sources $\mathbf{S}(t)$ en fonction des mesures :

$$\hat{\mathbf{S}}(t) = \mathbf{W}\mathbf{M}(t)$$

\mathbf{W} joue donc le rôle de l'inverse de \mathbf{A} (qui est inconnu).

La méthode la plus connue permettant de résoudre ce problème est l'analyse en composantes indépendantes (ou *ICA*) [Hyvärinen et Oja, 2000]. L'ICA suppose que les sources sont statistiquement indépendantes et cherche à décomposer les mesures en N_S sources les plus indépendantes possible.

De nombreux algorithmes existent pour effectuer une ICA. L'algorithme SOBI (blind source separation using second-order statistics) [Belouchrani *et al.*, 1997] est par exemple basé sur la diagonalisation conjointe de matrices de covariances spatiales. Les algorithmes les plus utilisés dans le domaine des ICM sont FastICA [Hyvärinen *et al.*, 2001] et INFOMAX [Lee *et al.*, 1999]. L'algorithme INFOMAX fut utilisé avec succès sur l'un des jeux de données de la *BCI competition III* [Blankertz *et al.*, 2006].

L'ICA est régulièrement utilisé dans les BCI en pré-traitement pour accroître le rapport signal sur bruit des données, ce qui peut permettre d'augmenter la précision de l'ICM [Qin *et al.*, 2005]. Dans [Kachenoura *et al.*, 2008] plusieurs algorithmes utilisant l'ICA sont passés en revue et comparés sur des jeux de données simulées. Dans [Naeem *et al.*, 2006], une comparaison est effectuée entre trois algorithmes utilisant l'ICA, un filtre laplacien et le CSP sur des données d'imagination motrice. La méthode CSP permet d'obtenir les meilleurs résultats, ce qui est dû à l'utilisation de l'appartenance aux classes. En effectuant le test sur une session différente de celle de l'apprentissage, les algorithmes basés sur l'ICA ne permettent pas de meilleurs résultats que l'emploi du filtre laplacien.

L'hypothèse suivant laquelle les sources corticales sont indépendantes entre elles est très discutable, cependant une grande partie du bruit extérieur est totalement indépendante de l'activité corticale. Ainsi l'ICA est souvent utilisée pour supprimer les artefacts liés au clignements des yeux ou aux contractions musculaires [Joyce *et al.*, 2004] : les composantes contenant les artefacts sont supprimées et le signal EEG est reconstruit.

3.3.2 Reconstruction de sources avec modélisation anatomique

Il est possible de réaliser un modèle anatomique de la tête du sujet et de s'en servir pour calculer la matrice de mélange. On parle alors de matrice de gain que l'on notera $\mathbf{G} \in \mathbb{R}^{N_E \times N_S}$. Les modèles anatomiques les plus courants utilisent trois couches pour modéliser le scalp, le crâne et l'intérieur du crâne en fonction d'une IRM anatomique du patient.

Il s'agit ensuite de calculer pour l'activation de chaque source corticale, modélisée par un dipôle, sa contribution au champ électro-magnétique sur les capteurs (qui est représenté par une colonne de la matrice de gain $\mathbf{g} \in \mathbb{R}^{N_E}$). C'est ce que l'on appelle le problème direct [Baillet *et al.*, 2001].

Les mesures sont modélisées comme la somme de la contribution de toutes les sources, auxquelles on ajoute très souvent un modèle de bruit gaussien :

$$\mathbf{M}(t) = \mathbf{GS}(t) + \mathbf{N}(t)$$

Il reste alors à résoudre un problème inverse pour déterminer l'activité des sources au cours du temps en fonction des mesures. Le nombre de sources pouvant être supérieur au nombre de capteurs, ce problème est souvent mal conditionné et il faut utiliser des algorithmes sophistiqués pour le résoudre.

Un problème majeur limite grandement l'utilisation de la reconstruction de sources pour les ICM : la nécessité de pouvoir effectuer la reconstruction de l'activité des sources en temps réel et *a fortiori*, sur des essais uniques. Les mesures effectuées en EEG étant très bruitées, on travaille normalement sur des données moyennées sur de très nombreux essais afin de rendre les résultats plus fiables, voire simplement exploitables. C'est sûrement la raison

pour laquelle la résolution du problème inverse n'a été utilisée que très récemment pour essayer d'améliorer le fonctionnement des ICM [Qin *et al.*, 2004, Grave de Peralta Menendez *et al.*, 2005, Congedo *et al.*, 2006].

La résolution du problème inverse ne peut pas créer de l'information aidant à la classification, au sens où toute celle-ci est déjà contenue dans les signaux mesurés sur les capteurs. Cependant la reconstruction de l'activité corticale ayant une base physiologique, la résolution du problème inverse peut aider à débruiter cette information et la rendre plus accessible [Grave de Peralta Menendez *et al.*, 2005, Babiloni *et al.*, 2007, Lotte *et al.*, 2008a].

Concernant les sources corticales, deux modélisations d'approche totalement différentes peuvent être utilisées.

- **Approche de type localisation** : On suppose qu'un nombre de sources très restreint (fixé ou à estimer) est à l'origine du signal mesuré ($N_S < 10 < N_E$). Il s'agit alors de déterminer la localisation de ces sources dans le cortex puis d'estimer leur activité.
- **Modèle de sources distribuées** : On suppose que l'ensemble des aires corticales contribuent au signal mesuré. On modélise donc l'activité de l'ensemble du cortex par un très grand nombre de sources spatialement distribuées dont il faut estimer l'activité. On utilise entre plusieurs centaines et plusieurs dizaines de milliers de sources, donc $N_S \gg N_E$.

Approche de type localisation L'approche de type localisation est appropriée quand un phénomène marqué modifie fortement l'activité cérébrale. Elle est par exemple particulièrement adaptée pour localiser des foyers épileptiques. Son utilisation en essai par essai pour les ICM (et en particulier celles basées sur l'imagination motrice) peut donc paraître peu adaptée.

Elle a pourtant été utilisée par [Qin *et al.*, 2004, Kamousi *et al.*, 2005] pour classifier l'imagination de mouvements des mains. Durant l'imagination de mouvements de la main droite, une diminution des rythmes μ et β (l'ERD) se produit dans la zone sensorimotrice de la main droite se trouvant dans l'hémisphère gauche. Par contraste, l'activité dans l'hémisphère droit paraît plus élevée et inversement. Les auteurs modélisent l'activité cérébrale par deux dipôles. Déterminer l'hémisphère ayant le plus d'activité (celui dans lequel se trouvent les deux, ou le plus grand dipôle) permet ainsi de classifier les essais. La modélisation de l'activité cérébrale sert ainsi directement à classifier les signaux.

Modèle de sources distribuées En se restreignant à des méthodes linéaires, l'approche des *sources distribuées* permet de reconstruire en temps réel une approximation de l'ensemble de l'activité corticale. Ce modèle paraît donc particulièrement adapté au cas des ICM.

Plusieurs études montrent que la reconstruction de l'activité corticale précédant l'extraction des features et la classification peut

permettre d'augmenter les taux de détection par comparaison aux mêmes traitements réalisés sur les données brutes des capteurs [Grave de Peralta Menendez *et al.*, 2005, Babiloni *et al.*, 2007, Noirhomme *et al.*, 2008, Besserve *et al.*, 2011, Ahn *et al.*, 2012]. Dans [Grosse-Wentrup *et al.*, 2009] les auteurs remarquent que ce gain est significatif pour les sujets dont de nombreux artefacts sont présents dans les données. Ceci peut être expliqué par le fait que certaines méthodes de reconstruction de sources¹⁰ permettent de séparer le bruit de l'activité corticale.

Cependant ces études utilisent généralement des traitements simples quand il s'agit de travailler directement sur les électrodes, rendant difficile la mesure d'un réel gain lié à la résolution du problème inverse. Dans [Ahn *et al.*, 2012] les auteurs testent une méthode de résolution du problème inverse sur un jeu de données de la *BCI competition* mais n'obtiennent qu'un taux de classification de 75,5% alors que les gagnants étaient arrivés à 94%. De même dans [Noirhomme *et al.*, 2008], bien que l'utilisation de la reconstruction de sources permette un gain important comparé à leur méthode, elle reste légèrement moins performante que la méthode de référence. Par contre dans [Lotte *et al.*, 2008a], les auteurs arrivent à améliorer très légèrement les performances (entre 1 et 2%) grâce à la résolution du problème inverse comparé au gagnant de la compétition.

Précision du modèle anatomique Contrairement à la MEG qui mesure des signaux magnétiques très peu sensibles à la traversée des milieux de conductivités différentes, la précision de la reconstruction de l'activité corticale basée sur l'EEG dépend fortement du modèle physique utilisé pour déterminer la matrice de gain \mathbf{G} . En effet, le crâne ayant une conductivité environ 30 fois inférieure à celle du scalp et du cerveau, il est préférable d'utiliser une bonne modélisation de ces différents tissus.

La plupart des modèles directs appliqués à l'EEG sont basés sur un modèle à 3 couches :

1. le scalp,
2. le crâne¹¹,
3. le liquide céphalo-rachidien et le cerveau.

Cependant trois niveaux de détails peuvent être utilisés pour modéliser ces couches :

1. Le modèle le plus grossier est constitué de trois sphères concentriques. Il a comme avantage sa simplicité et permet le calcul d'une solution

¹⁰Il s'agit des méthode de reconstruction de sources par filtrage de type Beamformer. Voir la section 3.2 du chapitre V page 146.

¹¹Le crâne ayant une conductivité faible et inhomogène, sa modélisation est particulièrement importante.

analytique. Il permet aussi de modéliser grossièrement l'inhomogénéité du crâne. Ce type de modèle est habituellement utilisé en MEG et non en EEG car il ne permet pas une localisation suffisamment précise. Il a tout de même été utilisé dans plusieurs études portant sur les ICM [**Kamoussi *et al.*, 2005**, **Noirhomme *et al.*, 2008**, **Lotte *et al.*, 2008a**, **Grosse-Wentrup *et al.*, 2009**, **Ahn *et al.*, 2012**].

2. Un modèle commun d'un crâne et d'un cerveau moyen peut être utilisé sans qu'il soit adapté au sujet. Plusieurs études en ICM sont ainsi basées sur le cerveau moyen de Montréal [**Grave de Peralta Menendez *et al.*, 2005**].
3. Pour obtenir des modèles plus adaptés aux sujets il est nécessaire de construire le modèle anatomique à partir d'une image IRM du patient. Quelques études ont ainsi été réalisées pour améliorer le fonctionnement d'ICM [**Babiloni *et al.*, 2007**, **Besserve *et al.*, 2011**]

L'équipe Athéna ayant développé de nombreux outils pour résoudre le problème direct, un des objectifs de cette thèse est l'exploitation de modèles anatomiques précis pour les ICM. Le chapitre V de ce manuscrit expose le travail effectué sur ce sujet.

Imagination motrice et rythmes sensori-moteurs

LE fait de réaliser des mouvements, ou simplement de les imaginer, engendre des variations des rythmes μ et β dans le cortex sensori-moteur. Ces variations peuvent être mesurées en EEG et servir pour réaliser des ICM. Ce chapitre décrit les travaux que nous avons effectués sur ce type d'ICM.

La première partie concerne l'étude de mouvements imaginés longs et soutenus. La seconde décrit un bouton commandé par la pensée qui est activé par le rebond dans la bande de fréquence β suivant l'imagination d'un mouvement. La dernière partie est consacrée à l'étude plus détaillée de ce rebond et plus généralement des modulations des rythmes μ et β engendrées par l'imagination de mouvements brefs.

Sommaire

1	Classification de mouvements continus	69
1.1	Enregistrements EEG durant des mouvements continus	69
1.1.1	<i>Objectifs</i>	70
1.1.2	<i>Le protocole expérimental</i>	70
1.1.3	<i>Détails techniques de l'enregistrement EEG</i>	72
1.1.4	<i>Réalisation du protocole avec OpenViBE</i>	74
1.2	Traitement des données	75
1.2.1	<i>Pré-traitements et découpage en fenêtres</i>	75
1.2.2	<i>Extraction des features</i>	76
1.2.3	<i>Sélection des features</i>	77
1.2.4	<i>Classification</i>	78
1.3	Résultats	78
1.3.1	<i>Discrimination entre les différentes tâches</i>	78
1.3.2	<i>Influence de la taille des fenêtres sur le bitrate</i>	79
1.4	Discussion	80
2	ICM utilisant des mouvements brefs	82
2.1	Le rebond bêta	82
2.1.1	<i>Précisions sur l'ERD et l'ERS</i>	82
2.1.2	<i>Avantages et inconvénients du rebond bêta</i>	83
2.2	ICM de navigation spatiale basée sur le rebond bêta	84
2.2.1	<i>Principe de fonctionnement</i>	84
2.2.2	<i>Extraction du signal de contrôle</i>	85
2.2.3	<i>Module de navigation spatiale</i>	88
2.3	Résultats	89
2.3.1	<i>Contrôle de l'interface par mouvements réels et imaginés</i>	89
2.3.2	<i>Utilisation de l'ICM dans différents contextes</i>	90
2.3.3	<i>Problèmes rencontrés</i>	90
2.4	Perspectives	90
3	Étude détaillée des mouvements brefs	92
3.1	Objectifs	92
3.2	Enregistrements EEG de mouvements brefs	93
3.2.1	<i>Le protocole de stimulation</i>	93
3.2.2	<i>Détails des enregistrements</i>	97
3.3	Analyse temps-fréquence des données	97
3.3.1	<i>Vision globale simplifiée de l'ERD/ERS</i>	97
3.3.2	<i>L'ERD/ERS : un phénomène complexe et sujet dépendant</i>	100
3.3.3	<i>Faible variabilité inter-sessions</i>	105
3.4	Traitement hors-ligne des données	106

3.4.1	<i>Chaîne de traitement hors-ligne</i>	106
3.4.2	<i>Détection et discrimination de mouvements</i>	106
3.4.3	<i>Gain lié à l'utilisation de la période post-imagination</i>	108
3.5	Détection et discrimination en-ligne de deux tâches	108
3.6	Conclusion et Perspectives	109
3.6.1	<i>Analyse des résultats</i>	109
3.6.2	<i>Vers une ICM totalement asynchrone</i>	110
4	Conclusion	113

1 CLASSIFICATION DE MOUVEMENTS CONTINUS

Durant la première partie de cette thèse nous nous sommes intéressés aux ICM basées sur l'imagination continue de mouvements moteurs. L'objectif original est de réaliser une ICM similaire à celle développée dans l'équipe de J. Wolpaw¹, mais en remplaçant une partie de l'apprentissage normalement effectué par l'utilisateur par un transducteur adaptatif plus flexible. Il s'agit pour cela de définir une nouvelle chaîne de traitement plus complexe adaptée à l'utilisation asynchrone des modulations des rythmes sensori-moteurs.

Une des pistes pour atteindre ce but est de reconstruire l'activité du cortex à partir des mesures effectuées par EEG sur le scalp (ce point fait l'objet du chapitre V de ce manuscrit²). Le signal d'entrée étant alors plus précis et moins bruité, il devrait être possible d'obtenir de meilleures performances par rapport à un signal d'entrée mesuré sur le scalp.

La première partie de ce chapitre décrit les expériences d'enregistrements EEG que nous avons réalisées durant le début de ce travail de thèse et la chaîne de traitement que nous avons mise en place pour analyser les modulations des rythmes sensori-moteurs produits par des mouvements soutenus.

1.1 Enregistrements EEG durant des mouvements continus

¹Cette ICM permet à l'utilisateur de contrôler un curseur sur un écran d'ordinateur en 1, 2 et maintenant 3 dimensions [Wolpaw *et al.*, 1991, Wolpaw et McFarland, 2004, McFarland *et al.*, 2010]. Voir section 2.1 du chapitre II.

²Malgré mon travail, la rapidité des ICM n'excèdent toujours pas la célérité des mains sur un clavier. Vous avez donc sous les yeux le fruit d'un travail conjoint de ma pensée et de mes mains et par conséquent sus-nommé *manuscrit*

1.1.1 Objectifs

L'expérience consiste à enregistrer l'activité cérébrale par EEG de sujets pendant qu'ils réalisent ou imaginent différents mouvements. Les données seront ensuite analysées pour retrouver quels mouvements le sujet produisait. Le but de l'expérience est double :

- trouver des tâches d'imagination motrice qui permettent une bonne discrimination.
- comparer différentes chaînes de traitement y compris celles incluant une reconstruction de l'activité cérébrale.

Afin de permettre un apprentissage des caractéristiques des différents mouvements et une estimation fiable des différentes méthodes de traitement, il est important de réaliser un grand nombre d'enregistrements de chacun des mouvements. D'autre part, l'objectif *in fine* de ces chaînes de traitement étant d'être utilisées sur des données évoluant en continu, nous avons fait le choix d'enregistrer des mouvements continus soutenus pendant plusieurs secondes.

Ces longues périodes de mouvements seront par la suite découpées en fenêtres pour l'analyse.

Ce choix a deux avantages :

- Chaque mouvement étant découpé en plusieurs fenêtres, cela multiplie le nombre d'essais pour l'apprentissage.
- Cela rapproche l'analyse hors-ligne des traitements qui seront effectués en-ligne. En effet, pour l'ICM développée par l'équipe de D. McFarland, le déplacement du curseur s'effectue toutes les 50 ms basé sur les 400 dernières millisecondes de signal. Il est donc nécessaire pour contrôler le mouvement du curseur de pouvoir déterminer pour chaque fenêtre de signal à quel mouvement il correspond.

1.1.2 Le protocole expérimental

Premier protocole expérimental Chaque session d'enregistrement est constituée d'une succession d'essais regroupés en blocs séparés par des pauses. Chaque essai est arrangé de la façon suivante (voir figure III.1) :

- Une croix de fixation apparaît au milieu de l'écran durant 0.5 seconde.
- Une consigne sous la forme d'une image représentant un mouvement est affichée en périphérie de l'écran en sus de la croix de fixation et reste durant 8 secondes.
- L'écran reste noir durant 1,5 secondes avant le début de l'essai suivant.

On demande au sujet de rester aussi détendu que possible et d'éviter de bouger. Il doit garder le regard fixé sur la croix quand elle apparaît et réaliser le mouvement demandé (ou l'imaginer suivant l'expérience) dès que la consigne est affichée à l'écran et jusqu'à ce qu'elle disparaisse.

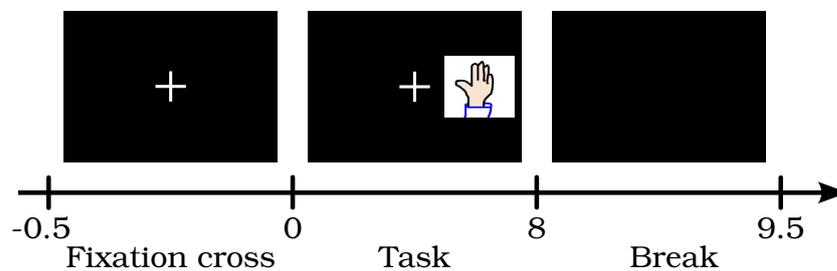


Fig. III.1 – Déroulement d'un essai pour l'enregistrement de l'imagination de mouvements soutenus. L'essai commence par l'apparition d'une croix de fixation. 0.5 secondes après, la consigne apparaît pour 8 secondes durant lesquelles le sujet doit imaginer le mouvement représenté. L'essai se termine par une pause de 1.5 secondes.

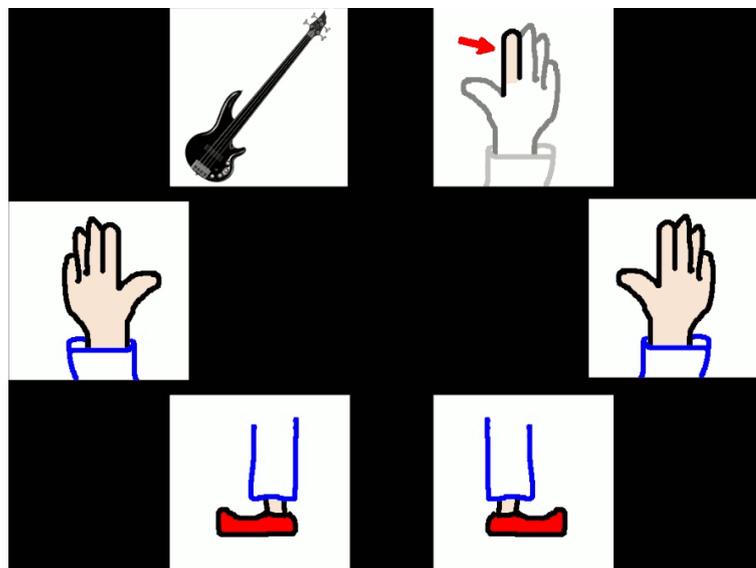


Fig. III.2 – Les 6 mouvements demandés au premier sujet, A1 : jouer des notes de guitare basse, bouger l'index droit, la main gauche, la droite, le pied gauche et le droit.

Les différentes tâches Durant la première expérience six différentes tâches d'imagination motrice ont été demandées au sujet (voir figure III.2) :

- mouvement de la main gauche
- mouvement de la main droite
- mouvement du pied gauche
- mouvement du pied droit
- mouvement de l'index droit
- imaginer jouer des notes de guitare basse (le sujet était un bassiste amateur)

Le but de cette première expérience était de voir si la reconstruction de l'activité cérébrale permettait de distinguer des tâches que l'on ne peut pas différencier en utilisant directement l'information provenant des capteurs.

Par la suite, le nombre de tâches a été réduit afin de permettre d'enregistrer plus d'essais pour chacun des mouvements. Nous avons alors retenu :

- mouvement de la main gauche
- mouvement de la main droite
- mouvement simultanés des deux pieds

Modification du protocole L'apparition de chaque consigne à une certaine position de l'écran³ rend le protocole plus intuitif. Cependant l'apparition des consignes entraîne une saccade oculaire du sujet vers l'image nouvellement affichée. Une détection de ces saccades pourrait ainsi permettre de déterminer le mouvement réalisé par le sujet. Afin de se prémunir de ce biais, nous avons modifié le protocole en affichant les consignes au centre de l'écran.

1.1.3 Détails techniques de l'enregistrement EEG

Disposition des électrodes Bien que l'information recherchée pour permettre la discrimination entre les différentes tâches se trouve au niveau des aires sensorimotrices, afin de permettre une reconstruction de l'activité cérébrale, 64 électrodes réparties suivant la nomenclature 10-20 (voir figure III.3) sont utilisées. De plus la position 3D de chacune des électrodes est enregistrée grâce à un système de localisation spatiale infrarouge avant et après l'expérience.

Enregistrement des données Les mesures sont amplifiées, échantillonnées (à 1024Hz) et numérisées grâce à un amplificateur 64 voies ANT. Deux ordinateurs sont utilisés :

- Un premier connecté à l'amplificateur sert uniquement à l'acquisition. Un programme faisant partie d'OpenViBE appelé "acquisition server" récupère par l'intermédiaire d'un driver⁴ les données provenant de l'amplificateur et

³La consigne demandant un mouvement de la main droite est affichée à droite de la croix de fixation selon les positions montrées sur la figure III.2.

⁴Ce "driver" est constitué de deux parties : un véritable driver permettant au système d'exploitation et aux programmes de communiquer avec l'amplificateur et un plug-in OpenViBE (aussi

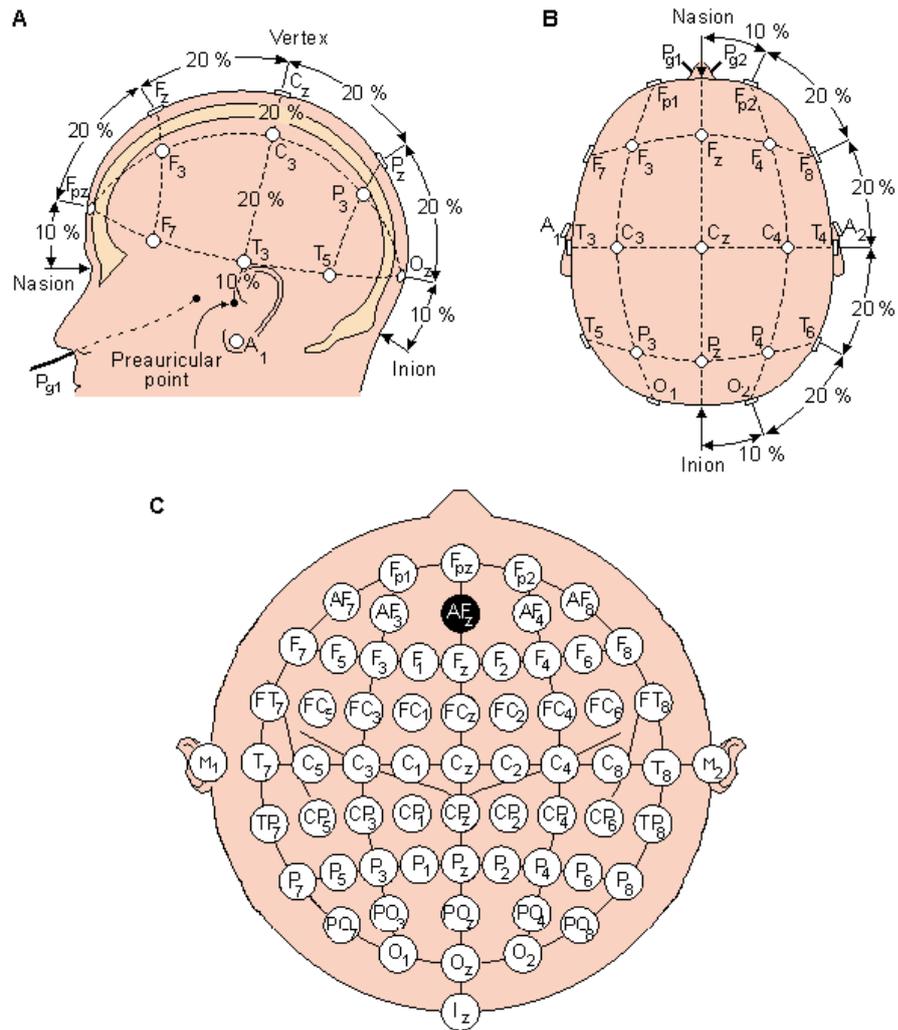


Fig. III.3 – Nom et positionnement des 64 électrodes suivant le système 10-20. L'électrode AFz est utilisée comme masse.

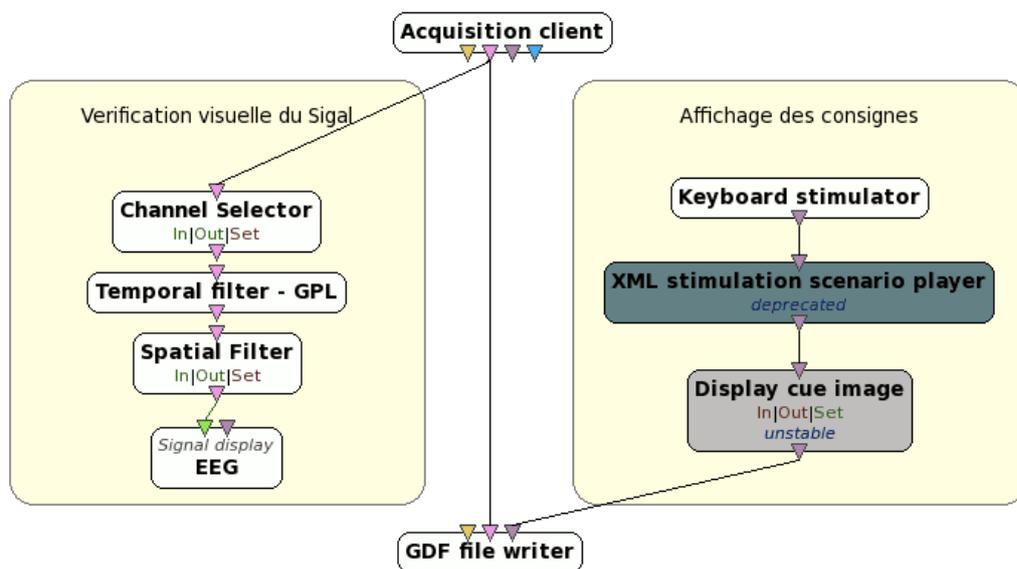


Fig. III.4 – Scénario OpenViBE pour l’enregistrement de mouvements moteurs continus.

les met sur le réseau à disposition des applications OpenViBE.

- Un deuxième ordinateur permet de faire tourner la partie d’OpenViBE appelée “designer” qui permet de traiter les données, de les enregistrer et de gérer les stimulations et feedbacks envoyés à l’utilisateur.

1.1.4 Réalisation du protocole avec OpenViBE

Les expériences en-ligne que nous avons effectuées ont été réalisées avec la plateforme OpenViBE, spécialement développée pour réaliser des ICM [Renard *et al.*, 2010]. Quelques détails sur son fonctionnement sont donnés dans le chapitre VI de ce manuscrit.

Le protocole étant destiné à l’analyse hors-ligne des résultats, aucun traitement n’est effectué en-ligne dans OpenViBE hormis la génération des consignes et une vérification visuelle de la validité des signaux enregistrés. Le scénario OpenViBE permettant d’effectuer l’expérience est représenté sur la figure III.4. Il est constitué de 4 parties.

1. L’acquisition des données : la boîte “Acquisition client” permet de se

appelé driver) permettant à OpenViBE d’utiliser le premier driver pour communiquer avec l’amplificateur. Le “driver” OpenViBE est facile à écrire. Le problème est qu’ANT ne fournissant pas de driver pour Linux, j’ai dû implémenter en C (puis finalement en C++ avec l’aide de Théodore Papadopoulos) un driver linux pour l’amplificateur à partir d’une documentation incomplète du protocole de communication de l’amplificateur.

connecter au serveur d'acquisition et de récupérer les données.

2. **L'affichage des données** : un traitement est effectué sur certaines électrodes afin de fournir à l'expérimentateur un aperçu des données acquises pour vérifier qu'il n'y pas d'incidents.
3. **L'affichage des consignes** :
 - la boîte “Keyboard stimulator” permet de démarrer la séquence de stimulation.
 - le timing et l'ordre des stimulations sont déterminés par un scénario écrit en XML et lu par la boîte “XML stimulation scénario player”⁵.
 - l'affichage de la croix de fixation et des consignes est effectué par la boîte “display cue image” que nous avons réalisée et qui a été intégrée dans une release d'OpenViBE.
4. **L'enregistrement des données** est effectué au format *gdf* par la boîte “GDF file writer”

1.2 Traitement des données

Le traitement des données est effectué hors-ligne grâce à la toolbox *MoffBCI* que nous avons développée. Son fonctionnement fait l'objet du chapitre VI de ce manuscrit.

1.2.1 Pré-traitements et découpage en fenêtres

Pré-filtrage Les données sont filtrées entre 4 et 40Hz afin de limiter l'influence de fréquences parasites, en particulier le courant continu à 50Hz.

Filtre spatial Quand on ne reconstruit pas l'activité cérébrale par résolution du problème inverse, un filtre laplacien spatial (voir figure II.1 page 58) est appliqué afin d'augmenter la résolution spatiale de l'EEG et ainsi la discrimination entre les tâches.

Découpage des essais en fenêtres Chaque essai correspond à la réalisation d'un mouvement durant 8 secondes. Afin d'être plus proche d'une analyse en-ligne, les essais sont découpés en fenêtres de 1 seconde toutes les 250 millisecondes (ce qui donne donc 29 fenêtres par essai). Pour chaque fenêtre, il s'agit de retrouver quel mouvement le sujet réalisait. La section 1.3.2 est consacrée à l'influence de la longueur des fenêtres sur le débit de l'information transmise.

⁵Cette boîte devrait être remplacée par le “Lua Stimulator” plus pratique et efficace mais qui ne fonctionne pas sur les ordinateurs de l'équipe.

1.2.2 Extraction des features

Les features⁶ utilisées sont la puissance dans les bandes μ et β du signal sur les différentes électrodes (ou sources corticales quand on reconstruit l'activité cérébrale). Plusieurs méthodes ont été essayées pour calculer la puissance du signal sur les différentes électrodes :

- par filtrage passe bande,
- par calcul du spectre par FFT ou par la méthode de Welch,
- par utilisation d'un modèle Auto-Régressif.

Nous avons aussi développé une méthode originale inspirée de la transformée en ondelettes.

Calcul des features par filtrage Une méthode fréquemment utilisée consiste à filtrer le signal par un filtre passe bande correspondant au μ ou au β , voire à une bande de fréquences englobant les deux [Li *et al.*, 2004].

Notons $m(t)$ la valeur du signal mesurée sur une électrode durant une fenêtre de temps $[t_1..t_2]$ dont l'on veut extraire les features. Soit $m_f(t) = H_f(m(t))$ le signal obtenu après filtrage dans la bande de fréquence f . Les features sont définies comme la moyenne quadratique du signal filtré sur chaque fenêtre :

$$F_f = \sum_{t=t_1}^{t_2} m_f(t)^2$$

Extraction des features par calcul du spectre fréquentiel Une autre méthode consiste à calculer le spectre fréquentiel par FFT sur chaque fenêtre puis de sélectionner comme features les fréquences d'intérêt [Pfurtscheller *et al.*, 1997].

Soit $[m_0..m_{n-1}]$ les valeurs du signal mesurées sur la fenêtre de temps $[t_1..t_2]$ dont l'on veut extraire les features. La transformée rapide de Fourier est définie par :

$$F_j = \sum_{k=0}^{n-1} m_k \omega_n^{jk}, \quad j = 0, \dots, n-1$$

avec $\omega_n = e^{-\frac{2\pi i}{n}}$ la n^{ieme} racine de l'unité.

Afin d'éviter les problèmes liés aux bordures des fenêtres, il est courant d'utiliser la méthode de Welch pour calculer le spectre [Welch, 1967] bien que la FFT soit aussi utilisée [Lal *et al.*, 2004].

Estimation du spectre par un modèle Auto-Régressif Quand la durée des fenêtres est réduite, il est possible d'utiliser un modèle Auto-Régressif pour calculer la puissance aux différentes bandes de fréquences [Schloegl *et al.*, 1997, McFarland *et al.*, 2006, McFarland et Wolpaw, 2008].

⁶Les features sont les valeurs caractéristiques du signal permettant la classification.

Méthode originale Nous avons développé une méthode originale adaptée à l'extraction de la puissance des rythmes dans l'EEG. Son fonctionnement est composé des étapes suivantes :

1. La fenêtre de signal dont on veut extraire les features est découpée en sous-fenêtres glissantes dont la largeur dépend de la fréquence.
2. Un fenêtrage en cloche est appliquée pour éviter les effets de bord.
3. Le spectrogramme du signal est calculé par transformée de Fourier sur chaque fenêtre.
4. La puissance retenue est la moyenne des puissances sur les différentes fenêtres.

Soient $m(t)$ la valeur du signal mesurée sur l'électrode dont l'on veut extraire la puissance à la fréquence f et F_{ech} la fréquence d'échantillonnage du signal. On définit par l_f la taille en nombre d'échantillons des sous-fenêtres glissantes. Pour permettre une estimation plus précise du spectre, la largeur des sous-fenêtres l_f dépend de la fréquence⁷. On calcule la puissance extraite de m à la fréquence f par :

$$F_f = \sum_{j=0}^{N_{sf}} \left| \sum_{k=0}^{l_f} m \left(j\alpha l_f + \frac{k}{F_{ech}} \right) e^{\frac{2\pi i f k}{F_{ech}}} h\left(\frac{k}{l_f}\right) \right|^2$$

où N_{sf} est le nombre de sous-fenêtres glissantes et αl_f le recouvrement entre sous-fenêtres voisines. Le fenêtrage en cloche est obtenu par la fonction h définie sur $[0, 1]$ par :

$$h\left(\frac{k}{l}\right) = 1 - \left(1 - 2\frac{k}{l}\right)^{10}.$$

Le fait de moyennner sur des sous-fenêtres glissantes permet d'obtenir une estimation, certes moins précise en terme de fréquence, mais plus robuste face au bruit important de l'EEG. Ceci est optimisé par l'adaptation de la taille des fenêtres en fonction de la fréquence.

1.2.3 Sélection des features

Il est possible d'utiliser plus d'électrodes que nécessaire, ou bien plusieurs bandes de fréquences pour l'extraction des features et d'utiliser uniquement les plus utiles pour la classification. Une étape de sélection des features les plus discriminantes est alors effectuée entre l'extraction des features et la classification. Quand on reconstruit l'activité corticale sous forme d'activité de milliers de sources, il est indispensable de sélectionner les features avant de procéder à la classification.

Notre algorithme de sélection des features est basé sur un t-test (le détail de cet algorithme est donné dans [Fruitet et al., 2010]). Sa particularité est que les

⁷ l_f est une fonction décroissante de f .

features sont sélectionnées de façon récursive ce qui permet d'éviter de sélectionner des features redondantes.

1.2.4 Classification

La classification est effectuée à l'aide d'un SVM [Vapnik, 1995, Vapnik, 1998] utilisant un noyau linéaire ou un noyau gaussien. C'est une méthode de classification couramment utilisée pour les ICM et qui donne de très bons résultats [Kaper *et al.*, 2004, Schlögl *et al.*, 2005].

Une étude des différentes méthodes de classification [Lotte *et al.*, 2007] souligne les avantages des SVM pour les ICM. Grâce à l'utilisation d'une pénalisation et d'une marge de séparation entre les classes, les SVM permettent généralement une bonne généralisation de l'ensemble d'apprentissage à l'ensemble de test et sont peu sensibles au problème de sur-apprentissage. Ils peuvent fonctionner avec un espace d'entrée de grande dimension, ce qui rend éventuellement moins nécessaire l'étape de sélection des features.

Le paramètre de régularisation λ (et la largeur du noyau gaussien σ) sont réglés de façon automatique par validation croisée ou double validation croisée (voir la section 3 du chapitre VI).

1.3 Résultats

1.3.1 Discrimination entre les différentes tâches

Les performances de discrimination entre les différentes tâches d'imagination motrice sont données pour deux sujets dans les tableaux III.1 et III.2. La reconstruction de l'activité cérébrale n'est pas utilisée.

Mouvement	basse	index D	main G	main D	pied G	pied D
basse	*	76%	64%	75%	72%	69%
index D	76%	*	56%	(47%)	74%	65%
main G	64%	56%	*	64%	55%	56%
main D	75%	(47%)	64%	*	57%	56%
pied G	72%	74%	55%	57%	*	(50%)
pied D	69%	65%	56%	56%	(50%)	*

Tab. III.1 – Taux de classification entre les six tâches d'imagination motrice pour le premier sujet (A1) basé sur des fenêtres de 1 seconde de signal. Tous les résultats sauf ceux entre parenthèses sont significativement différents du hasard ($p > 95\%$).

Pour le sujet A1, le fait de présenter six tâches différentes limite le nombre d'essais pour chaque tâche. La quantité de données disponibles pour effectuer

Mouvement	main G	main D	pieds
main G	*	69%	81%
main D	69%	*	75%
pieds	81%	75%	*

Tab. III.2 – Taux de classification entre les trois tâches d’imagination motrice pour le sujet A2 basé sur des fenêtres de 1 seconde de signal.

l’apprentissage et la validation des performances est alors extrêmement limitée. Le fait d’utiliser la validation croisée⁸ permet d’estimer un intervalle de confiance sur les résultats⁹. Ainsi pour la discrimination entre jouer de la basse et bouger l’index droit, qui sont les tâches apparaissant comme les plus faciles à distinguer, le taux de discrimination moyen est de 76% avec un intervalle de confiance à 95% de [70% – 81%]. Pour donner une idée de l’importance de cette variation cela correspond à un taux de transfert d’information¹⁰ de l’ICM variant entre 7 et 18 bit/min.

La variation des résultats ne permet pas de comparer précisément les performances des différentes tâches. Cependant, on peut déterminer deux résultats négatifs sur le pouvoir discriminant de l’EEG sur cette expérience. Il n’est pas possible de distinguer :

- l’imagination d’un mouvement de l’index droit de l’ensemble de la main droite,
- l’imagination d’un mouvement du pied droit de celui du gauche¹¹.

Pour ces raisons le nombre de tâches motrices des expériences suivantes a été revu à la baisse et l’imagination des deux pieds est effectuée de façon simultanée et non plus comme deux tâches différentes.

Pour le sujet A2, le couple de tâche permettant la meilleure discrimination est la main gauche versus les pieds III.2.

1.3.2 Influence de la taille des fenêtres sur l’information transmise

La figure III.5 représente l’impact de la taille des fenêtres choisies sur le taux de classification. Plus la fenêtre utilisée est grande, plus elle contient d’information, et plus le taux de classification augmente. Cependant, plus la fenêtre utilisée

⁸Il s’agit plus précisément d’une double validation croisée pour permettre d’ajuster plusieurs paramètres comme le nombre de features à sélectionner ou le terme de pénalisation pour le SVM.

⁹Il faut cependant supposer que les résultats obtenus sur les différents ensembles de test sont indépendants et qu’ils suivent une distribution gaussienne.

¹⁰Le taux de transfert d’information est défini dans la section suivante.

¹¹Cela est dû au fait que les aires motrices des pieds, bien que séparées sur les deux hémisphères, sont situées juste “en face” l’une de l’autre (voir figure I.24).

est grande, moins l'ICM est réactive et plus le délai entre l'imagination d'un mouvement et l'envoi de la commande est élevé.

Taux de transfert d'information (ou *bit-rate*) Pour mieux mesurer les performances d'une ICM il est possible de calculer le taux de transfert d'information. Soit N le nombre de choix offerts à l'utilisateur. Si la classification avait une réussite de 100%, chaque sélection permettrait de transmettre $\log_2(N)$ bits d'information.

Cependant, le choix de l'utilisateur n'est pas forcément celui sélectionné par l'ICM. La théorie de la communication de Shannon [Shannon et Weaver, 1964] permet de calculer l'information transmise malgré les erreurs de sélection. Si on note P la probabilité que le bon choix soit sélectionné, le nombre de bits d'information transmis par essai est de :

$$B = \log_2(N) + P \log_2(P) + (1 - P) \log_2\left(\frac{1 - P}{N - 1}\right)$$

Si on note T le temps nécessaire pour effectuer chaque choix, le taux de transfert d'information est donné par :

$$T_R = \frac{B}{T} \text{ bit/min}$$

Une discussion sur le taux de transfert d'information est effectuée dans [Wolpaw *et al.*, 2000]. L'attention est portée sur le fait que pour une ICM disposant de $N = 2$ choix, le fait de passer d'un taux de classification de 80% à 90% permet de doubler¹² le taux de transfert d'information.

La figure III.6 représente l'évolution du taux de transfert d'information en fonction de la taille des fenêtres choisies.

Contrairement au taux de classification qui croît avec la longueur des fenêtres, le taux de transfert d'information varie de façon plus complexe :

- Pour une fenêtre trop courte (300ms) l'information transmise est faible car il y a trop peu de données pour obtenir une bonne classification.
- Pour des fenêtres comprises entre 0.5 et 1.3s, le taux de transfert d'information reste stable. En effet, augmenter la durée des fenêtres permet une meilleure classification mais consomme davantage de temps.
- Pour des fenêtres plus longues, l'allongement de la durée des essais n'est plus suffisamment compensé par une amélioration des performances de classification.

1.4 Discussion

Bien que le protocole expérimental ait été établi pour ressembler à une utilisation en-ligne d'une ICM basée sur l'imagination motrice, il existe cependant

¹²plus précisément cela augmente le taux de transfert d'information de 90%.

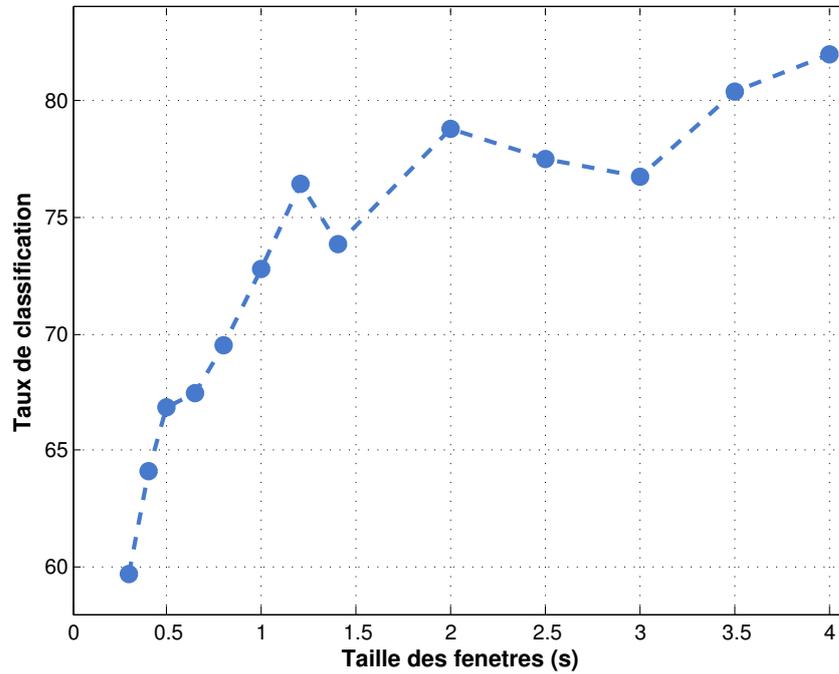


Fig. III.5 – Évolution du taux de classification entre la main droite et les pieds pour le sujet A2 en fonction de la taille des fenêtres.

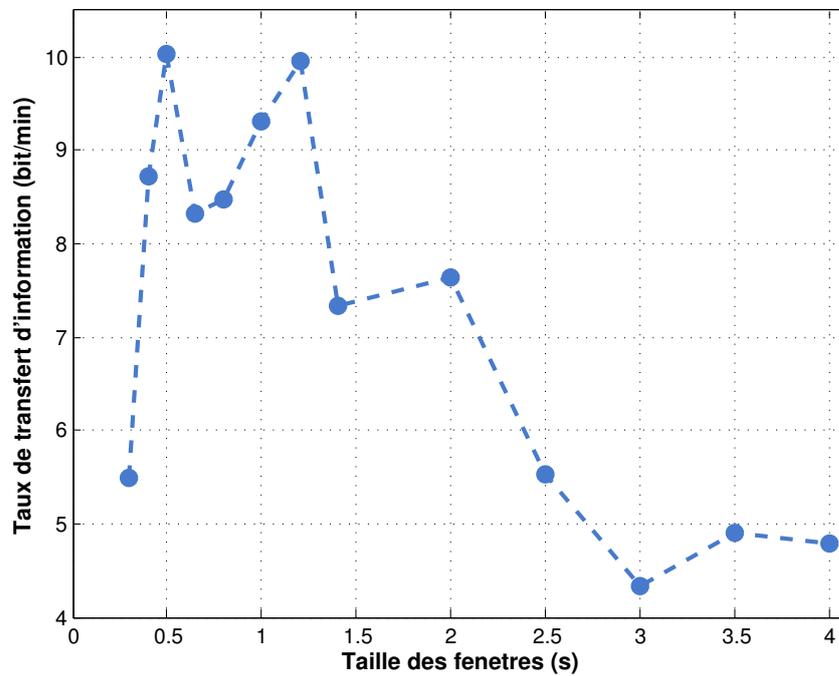


Fig. III.6 – Évolution des taux de transfert d'information en fonction de la taille des fenêtres pour un transducteur qui utiliserait la distinction entre la main droite et les pieds (pour le sujet A2).

de fortes différences. Tout d'abord, pour le sujet : non seulement il ne reçoit aucun feedback sur l'interprétation des commandes qu'il est supposé envoyer, mais surtout les tâches qu'il doit accomplir sont peu réalistes. En effet, le but d'une ICM basée sur de l'imagination continue est de pouvoir obtenir un contrôle multi-dimensionnel continu. Or, pour ce premier protocole le sujet ne doit qu'imaginer des périodes de 8 secondes d'un même mouvement. Il n'y a donc pas de modulation en continu, ni de passage d'un mouvement à un autre. Il est donc probable qu'une telle différence de contexte influe sur la nature des signaux.

Concernant l'absence de feedback, il paraît raisonnable de penser qu'une fenêtre extraite durant un mouvement de la main durant 8 secondes serait similaire au signal obtenu durant une réelle utilisation en-ligne d'une telle ICM, même si cela reste à démontrer. Il serait d'ailleurs intéressant de vérifier si un apprentissage effectué sur un tel jeu de données (c'est à dire obtenu par un protocole sans feedback et avec des mouvements continus essais par essais) pourrait être utilisé pour ensuite contrôler de façon continue une ICM. Il est probable qu'il est préférable d'utiliser cette base de donnée hors-ligne pour établir un premier apprentissage et une calibration des paramètres et d'ensuite l'affiner durant l'utilisation en-ligne.

La calibration des paramètres ou sélection des méthodes à utiliser peut aussi être effectuée hors-ligne en re-simulant une analyse en-ligne. Dans [Fruitet *et al.*, 2010], les auteurs simulent le mouvement d'un curseur hors-ligne basé sur des enregistrements où les utilisateurs contrôlaient un curseur en-ligne.

2 ICM UTILISANT DES MOUVEMENTS BREFS : LES AVANTAGES DU REBOND BÊTA

2.1 Le rebond bêta

2.1.1 Précisions sur l'ERD et l'ERS

Comme expliqué dans le 1^{er} chapitre (section 4.2.2 page 40), la réalisation ou l'imagination d'un mouvement produit généralement une baisse des oscillations dans le μ et le β durant le mouvement (ce qu'on appelle ERD, pour *Event related Desynchronisation*) suivi d'une augmentation supérieure à la normale une fois le mouvement terminé (appelé ERS, pour *Event related Synchronisation*)¹³. Cette augmentation suivant le mouvement est particulièrement visible dans la bande de fréquences β et l'on parle parfois de rebond β pour la désigner.

¹³La troisième partie de ce chapitre est consacrée à l'étude de l'imagination de mouvements brefs. On y montrera que le phénomène d'ERD/ERS peut être complexe et variable suivant les sujets.

Bien que de nombreuses ICM soient décrites comme utilisant l'ERD/ERS comme principe biologique à l'origine du contrôle de l'interface [Wolpaw *et al.*, 2000], très peu d'entre elles utilisent réellement l'ERS. En effet, la plupart des ICM basées sur l'imagination motrice utilisent la modulation des rythmes μ et β produite durant le mouvement et parlent de façon trop générale d'ERD/ERS, alors qu'il ne s'agit généralement que d'ERD, l'ERS ayant lieu majoritairement après le mouvement et n'étant donc pas utilisé.

Cas particulier

*Certaines ICM sont basées sur la modulation en continu des rythmes moteurs (comme celle développée au Wadsworth Center [McFarland *et al.*, 2010]). Après un long apprentissage (basé originellement sur l'imagination motrice) les utilisateurs peuvent augmenter et diminuer la puissance des rythmes μ et β . Il s'agit donc de synchronisation et de désynchronisation, mais il est plus difficile de faire un lien avec un événement et donc de parler d'ERD/ERS.*

Pourtant l'ERS peut aussi être utilisé pour le contrôle d'ICM. En effet certaines études montrent que l'ERS dans la bande β permet de détecter l'imagination de mouvements brefs des pieds [Pfurtscheller *et al.*, 1996] mais aussi d'autres membres [Pfurtscheller *et al.*, 2005]. Ceci peut permettre de réaliser des ICM basées uniquement sur le rebond β [Lotte *et al.*, 2008b, Pfurtscheller et Solis-Escalante, 2009, Müller-Putz *et al.*, 2010] ou sur l'ERD et l'ERS des rythmes μ et β [Bai *et al.*, 2008, Fruitet *et al.*, 2011].

2.1.2 Avantages et inconvénients du rebond bêta

Le problème quand on veut utiliser le rebond β ou plus généralement l'ERS, est que la synchronisation n'intervient qu'après la fin du mouvement. Quand le sujet désire envoyer une commande (que ce soit spontanément ou en réponse à un stimulus¹⁴), celle-ci ne peut avoir lieu qu'après la fenêtre de détection du rebond (environ 2 secondes après la fin du mouvement). Il y a donc un délai de quelques secondes entre le moment où l'utilisateur initie l'envoi de sa commande et le temps où elle peut être reconnue. L'allongement de la période nécessaire à l'envoi d'une commande réduit aussi le taux de transfert d'information (voir figures III.5 et III.6).

Cependant, le rebond β a l'avantage d'être clairement distinguable de l'état de repos. Ainsi il est possible de réaliser facilement, et avec des périodes d'apprentis-

¹⁴Petit quiz : Si la commande est envoyée spontanément il s'agit d'un protocole asynchrone. Vrai ou Faux ? (si vous avez répondu faux allez lire la page 19)

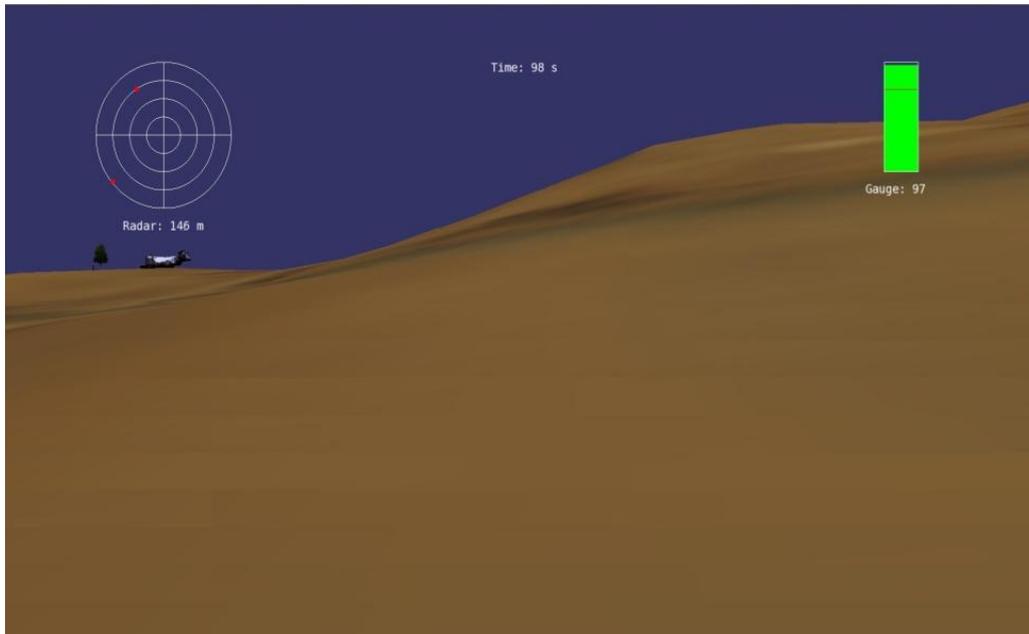


Fig. III.7 – Environnement virtuel simplifié utilisé en dehors de la salle immersive. La cible est constituée d’une vache que le sujet doit atteindre. En haut à gauche est affiché un radar permettant à l’utilisateur de repérer la cible quand elle n’est pas dans son champ de vision. En haut à droite une jauge donne des indications sur la puissance mesurée dans la bande de fréquences β .

sage courtes, des ICM basées sur la distinction du rebond β de l’activité de repos [Pfurtscheller et Solis-Escalante, 2009, Müller-Putz *et al.*, 2010].

2.2 ICM de navigation spatiale basée sur le rebond bêta

Durant cette thèse, j’ai participé à l’encadrement d’un ingénieur pour la réalisation d’une ICM permettant la navigation spatiale dans un environnement immersif. Le but est d’utiliser une ICM de type bouton commandé par la pensée pour interagir avec la salle de réalité virtuelle présente à l’INRIA Sophia-Antipolis.

2.2.1 Principe de fonctionnement

Le choix de l’ICM s’est porté sur l’adaptation d’une ICM existant dans OpenViBE, appelée *Tie-fighter*. Cette ICM permet à l’utilisateur de soulever un *Tie-fighter*¹⁵ grâce au rebond β généré à la suite d’un mouvement des pieds réel ou imaginé [Lotte *et al.*, 2008b].

¹⁵vaisseau spatial de l’empire dans *Star Wars*

Le signal de contrôle On utilise la variation de la puissance dans la bande de fréquence β (16-24Hz). Quand l'utilisateur effectue un mouvement des pieds (réel ou imaginé) cela engendre un ERS environ 1 seconde après le mouvement. C'est cette augmentation de puissance dans la bande de fréquence β qui déclenche l'envoi du signal de contrôle.

L'utilisateur dispose donc d'un bouton qu'il peut "actionner" en imaginant un bref mouvement des pieds.

Du signal de contrôle à la navigation spatiale Le signal de contrôle étant extrêmement pauvre, la particularité de cette ICM réside dans l'utilisation de ce simple bouton pour naviguer dans un monde immersif.

La navigation s'effectue par deux types de déplacements qui alternent à chaque rebond β :

- une rotation dans le sens horaire.
- une translation vers l'avant.

L'utilisateur initie la communication par un rebond β , l'avatar contrôlé par l'utilisateur commence alors à effectuer une rotation sur lui-même dans le sens horaire. Quand il est face à la direction désirée, l'utilisateur arrête la rotation par un rebond β ¹⁶. L'avatar se met alors à avancer tout droit. Quand l'utilisateur veut changer de direction, il lui suffit de produire un nouveau rebond β et l'avatar se remet à tourner et ainsi de suite. La figure III.10 représente l'automate à états finis utilisé pour la navigation.

Tâche de navigation spatiale Afin de vérifier si le sujet est capable d'utiliser l'ICM, une cible est placée dans l'environnement immersif et l'on demande au sujet de l'atteindre. La figure III.7 représente l'environnement virtuel utilisé, un environnement plus complexe et plus détaillé peut cependant être utilisé.

2.2.2 Extraction du signal de contrôle

La reconnaissance des rebonds bêta est effectuée à l'aide du scénario OpenViBE représenté sur la figure III.8 et détaillé ci-dessous.

Acquisition des données L'acquisition se fait toujours grâce à la boîte "Acquisition client" qui se connecte par TCP/IP à "l'acquisition server". Les données sont enregistrées sur Cz, C3 ou C4 suivant si les pieds, la main droite ou la gauche sont utilisés pour contrôler l'ICM, ainsi que sur 4 électrodes voisines pour réaliser un filtrage spatial. Deux amplificateurs peuvent être utilisés :

- l'amplificateur ANT 64 voies, la fréquence d'échantillonnage est alors fixée à 512Hz ;

¹⁶Afin de produire un rebond β au moment t désiré l'utilisateur doit anticiper et effectuer un mouvement se terminant environ 1 seconde avant l'instant t .

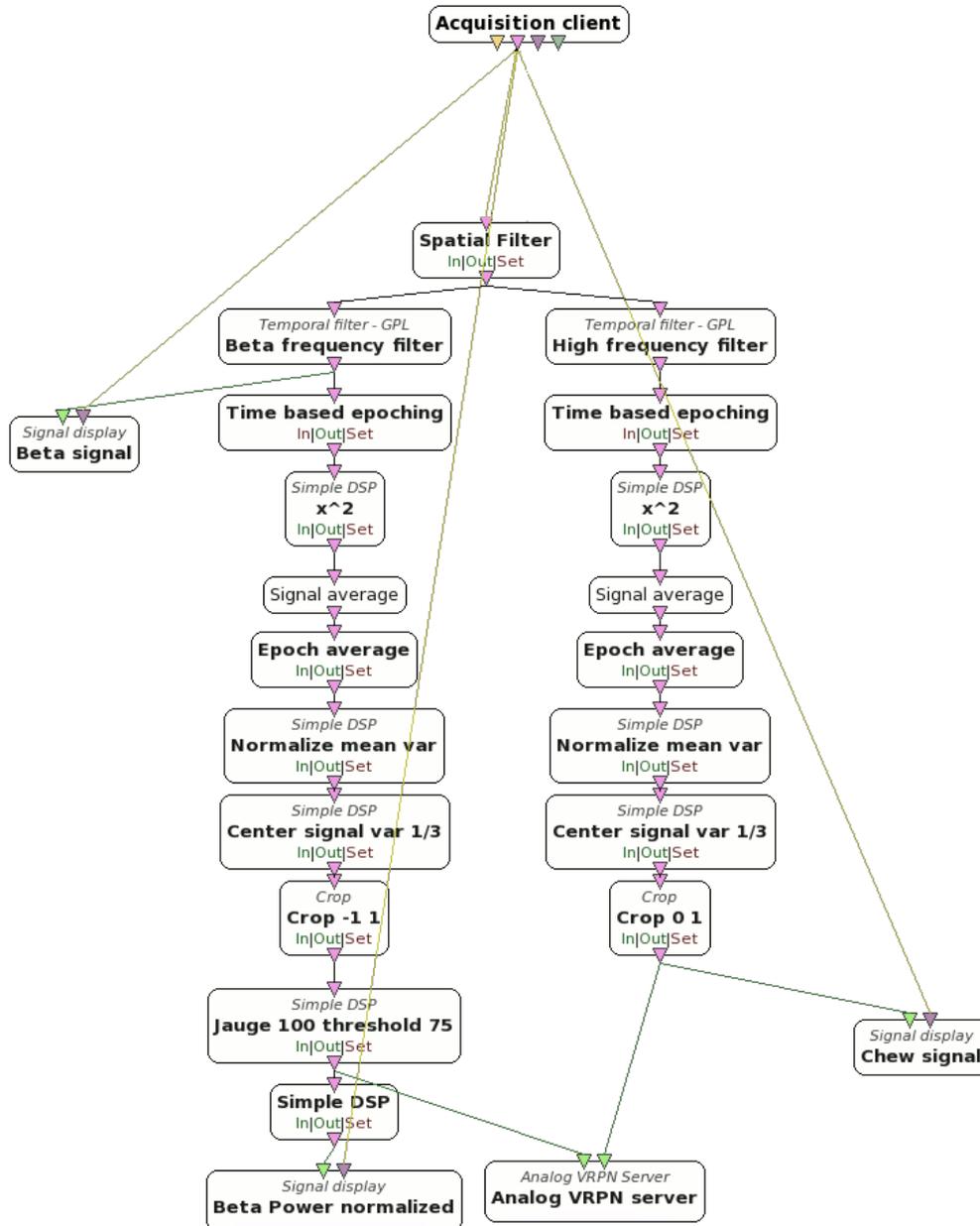


Fig. III.8 – Scénario OpenViBE pour la détection des rebonds β .

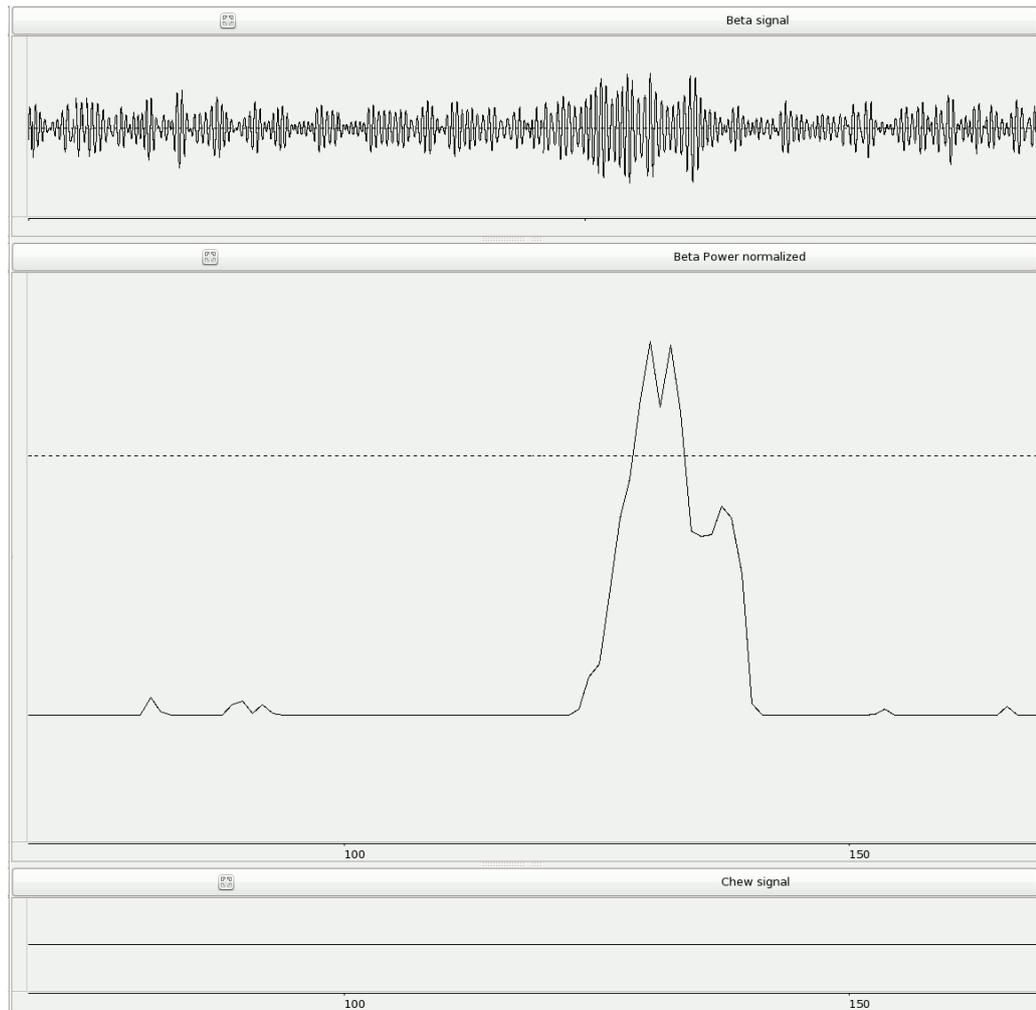


Fig. III.9 – Le rebond β . En haut : signal après filtrage dans la bande de fréquence β . Au milieu : le signal servant à la détection des rebond β . Le seuil est représenté en pointillés. En bas : signal de détection des serremments de mâchoire.

- un amplificateur mobile bluetooth à 8 voies : le *Gtec gMobilab+*. La fréquence d'échantillonnage est alors limitée à 256Hz.

Filtrage spatial Afin d'améliorer la précision des signaux, un filtrage spatial de type laplacien est ajouté au scénario original.

Extraction de la puissance bêta Les données sont filtrées entre 16 et 24Hz, puis la puissance est calculée par la boîte "Simple DSP". Une moyenne est calculée par fenêtres glissantes grâce aux boîtes "Time based epoching" et "Signal average" afin de débruiter le signal de contrôle.

Normalisation du signal de contrôle Le signal est ensuite normalisé grâce à la moyenne et l'écart type calculés durant une période de repos de l'utilisateur. Pour cela une période de repos de 30 secondes est enregistrée au début de l'expérience.

Transmission du signal de contrôle Le signal de contrôle est ensuite envoyé à l'application de navigation spatiale par un protocole VRPN¹⁷.

Détection des serremments de mâchoires Les serremments de mâchoires engendrant une augmentation de puissance importante dans toutes les fréquences, ils risquent d'être interprétés comme des rebonds β . Une chaîne de traitement identique, mais avec un filtre passe haut à 16Hz au lieu du filtrage de la bande bêta, est donc utilisé pour les détecter et les supprimer.

2.2.3 Module de navigation spatiale

Le module de navigation spatiale est une application indépendante d'Open-ViBE écrite en C++ par Nicolas Servant. Elle permet le déplacement dans un monde virtuel et l'affichage sur un écran standard ou dans la salle immersive de l'INRIA Sophia Antipolis¹⁸. Elle prend en entrée un signal de contrôle fourni par une ICM ou par une interface homme-machine traditionnelle comme une manette ou un clavier.

Ces entrées peuvent être paramétrées pour contrôler le mouvement de différentes façons : par l'association simple d'un bouton à une direction de rotation ou de translation, ou par un mécanisme plus complexe, basé sur un automate à états finis. L'utilisation d'un automate permet avec un faible nombre d'entrées d'obtenir un contrôle complexe du mouvement. La figure III.10 représente l'automate utilisé pour le contrôle de cette ICM.

¹⁷Virtual-Reality Peripheral Network

¹⁸Les espaces immersifs des différents centres de l'INRIA ne sont pas pour l'instant compatibles entre eux.

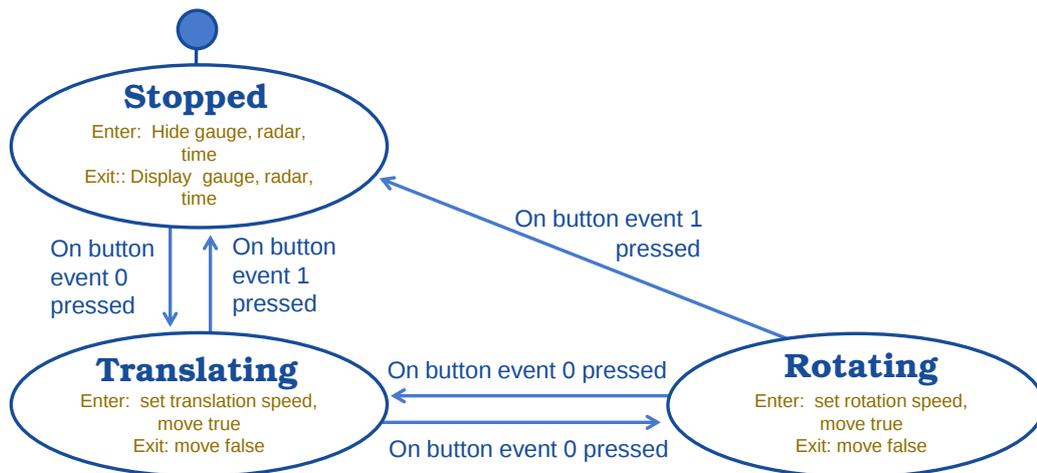


Fig. III.10 – Automate utilisé pour la navigation spatiale basée sur le rebond β . La détection d'un rebond β correspond à l'appui et au relâchement du bouton 0. Le bouton 1 correspond à l'atteinte de la cible.

2.3 Résultats

Difficultés pour la mesure des performances Cette ICM a pour l'instant été utilisée sur plusieurs sujets et dans différents contextes mais sans l'élaboration d'un protocole rigoureux permettant une mesure précise des performances.

Il s'agit d'une ICM asynchrone de type bouton contrôlé par la pensée. Le sujet est relativement libre pour réaliser la tâche demandée (c'est à dire atteindre la cible dans l'environnement virtuel). On ne sait donc pas quand le sujet tente d'envoyer les commandes. Les seules mesures possibles sont une évaluation globale des performances ou une évaluation subjective.

Avant de réaliser un protocole permettant une telle mesure il était nécessaire de procéder à des tests de façon empirique. Les résultats présentés sont le résumé des différentes expériences qui ont été réalisées et sont donc exprimés de façons qualitatives.

2.3.1 Contrôle de l'interface par mouvements réels et imaginés

Tous les sujets ayant utilisé l'ICM ont réussi à obtenir un contrôle plus que correct à l'aide de mouvements réels. Le contrôle à l'aide de mouvements imaginés est plus difficile et varie davantage d'un sujet à l'autre voire d'une session à l'autre.

D'autre part, le système de calibration est basé sur une période de référence de 30 secondes durant laquelle le sujet ne doit effectuer aucun mouvement. Or, il est fréquent que la période de référence conduise à un seuil de détection mal ajusté.

- S'il est trop haut, très peu de rebonds bêta sont détectés.

- S’il est trop bas, trop de faux positifs rendent le contrôle de l’interface impossible.

Une solution pour améliorer la gestion du seuil serait d’utiliser, en plus de la période repos, une période durant laquelle l’utilisateur effectue des mouvements. Ainsi le seuil pourrait être ajusté en fonction de la puissance moyenne du signal, mais aussi en fonction de l’amplitude des rebonds.

2.3.2 Utilisation de l’ICM dans différents contextes

L’ICM a d’abord été utilisée avec succès dans le laboratoire EEG sur un écran 2D. Des tests ont ensuite été effectués dans l’espace immersif en 2 puis 3 dimensions. L’absence de cage de Faraday et l’importance des champs électromagnétiques générés par les installations de réalité virtuelle n’ont pas semblé influencer le contrôle de l’interface.

2.3.3 Problèmes rencontrés

En sus de multiples problèmes techniques rencontrés, deux problèmes liés au fonctionnement de l’ICM nous ont obligé à adapter son fonctionnement.

1. **La présence de rebonds multiples dans la bande β** : pour certains sujets, l’ERS est caractérisé par plusieurs bouffées d’oscillations β après le mouvement générant plusieurs pics de puissance dépassant le seuil. Chaque mouvement générant alors l’envoi de plusieurs commandes au lieu de la seule escomptée. Le problème a été résolu par l’introduction d’une période réfractaire suivant chaque rebond β durant laquelle aucune commande ne peut être envoyée.
2. **Difficulté d’utiliser l’imagination motrice des pieds quand le sujet est debout.** D’après les expériences effectuées, il semblerait qu’il soit non seulement plus difficile au sujet d’imaginer réaliser un mouvement des pieds s’il est debout, mais que la puissance de l’ERS soit aussi grandement atténuée ce qui rend la détection des rebonds β plus difficile.

2.4 Perspectives

Mesure des performances Il serait nécessaire d’établir un protocole permettant une mesure quantitative des performances des différents sujets. D’autre part, il serait intéressant de voir si l’utilisation de l’espace immersif permet une meilleure motivation des sujets se traduisant par une augmentation des performances. Une amélioration des performances grâce à un protocole plus stimulant est rapporté dans [Leeb *et al.*, 2007]. Cependant, il n’est pas sûr qu’un tel gain soit suffisant

pour être mesurable au travers d'un nombre forcément réduit de sujets et d'expériences. Par contre, il est probable que l'espace immersif donne une impression subjective de meilleur contrôle de l'interface.

Amélioration de la détection des rebonds bêta La méthode de détection des rebonds β est pour l'instant très sommaire. La troisième partie de ce chapitre est constituée d'une étude plus détaillée des ERD/ERS afin de permettre une meilleure détection des mouvements imaginés par l'utilisateur. Cela permettra d'utiliser une méthode de détection plus complexe et ainsi de faciliter l'utilisation de l'interface.

Utilisation de plusieurs commandes Il est possible, en utilisant l'ERD et l'ERS, de détecter et de différencier plusieurs mouvements [Fruitet *et al.*, 2011] et ainsi d'utiliser un plus grand nombre de commandes. Une première amélioration serait de conserver l'imagination de mouvements des pieds pour avancer mais d'utiliser l'imagination de mouvements de la main droite ou gauche pour déterminer le sens de rotation quand on arrête d'avancer. Ainsi l'utilisateur n'aurait pas à faire un tour quasiment complet sur lui-même quand il désire tourner à gauche. Bien que trois commandes soient alors utilisées, suivant l'action que l'avatar réalise (marche ou rotation) seulement un ou deux signaux de contrôle seraient attendus. Il ne serait donc pas nécessaire d'être capable de différencier les mouvements des pieds de ceux des mains.

Réalisation d'une ICM hybride pour une utilisation ludique À cause du taux de transfert extrêmement faible de l'ICM, le mode de déplacement est lent et laborieux. Pourtant l'utilisation de l'ICM dans l'espace immersif, bien qu'un peu frustrante, reste amusante. Une idée pour rendre l'interface plus dynamique serait d'utiliser une combinaison de commandes envoyés par des interfaces traditionnelles et par ICM.

Il pourrait, par exemple, être intéressant d'utiliser l'orientation de la tête du sujet¹⁹ pour contrôler le déplacement de l'avatar et de réserver le signal de contrôle de l'ICM pour envoyer des commandes dont le *bitrate* est moins important. L'utilisateur pourrait par exemple se déplacer par lévitation dans un monde immersif où il devrait détruire des ennemis par la pensée (ou délivrer des petits chatons, pour les pacifistes convaincus).

¹⁹La position et l'orientation de la tête du sujet dans l'espace immersif sont connues grâce à un système de *head-tracking*.

3 ÉTUDE DÉTAILLÉE DES MOUVEMENTS BREFS : VERS UNE ICM ASYNCHRONE À COMMANDES DISCRÈTES

Contrairement aux modulations continues des rythmes moteurs, qui sont largement étudiées, l'utilisation de mouvements brefs pour contrôler une ICM est encore très peu répandue. Pourtant l'imagination de mouvements brefs paraît particulièrement adaptée pour la réalisation de nombreuses ICM. D'autre part, malgré la méthode de détection très sommaire des rebonds β , les premiers résultats de l'ICM utilisée pour la navigation spatiale furent encourageants. Pour toutes ces raisons, nous avons décidé d'approfondir notre étude des mouvements en nous concentrant sur les mouvements brefs.

3.1 Objectifs

Utiliser l'ERS pour détecter et distinguer différents mouvements brefs Quelques équipes s'intéressent à l'utilisation du rebond bêta produit par le mouvement des pieds pour réaliser un interrupteur commandé par la pensée (ou *brain-switch*) [Pfurtscheller et Solis-Escalante, 2009] ou pour commander une ICM [Bai *et al.*, 2008]. Pourtant, lorsque nous avons commencé cette étude, à notre connaissance, aucune autre équipe n'avait essayé d'utiliser l'ERS pour distinguer des mouvements brefs imaginés avec différentes parties du corps²⁰. Le premier objectif de cette étude est de montrer qu'il est possible d'utiliser le rebond β (et plus généralement l'ERS conjointement à l'ERD) pour non seulement distinguer différents mouvements d'une période de repos, mais aussi les uns des autres.

Améliorer la détection des rebonds bêta L'autre objectif de cette étude est de réaliser une méthode de détection des rebonds β plus performante que celle utilisée par l'ICM de navigation spatiale décrite dans la partie précédente. Il est en effet possible d'utiliser l'ERD et l'ERS dans les bandes de fréquences μ et β plutôt que de se restreindre au rebond β .

Réaliser une ICM asynchrone à commandes discrètes *In fine* le but est de permettre au sujet d'envoyer quand il le désire (ICM asynchrone) une commande parmi un ensemble de commandes possibles. Il s'agit donc de boutons commandés par la pensée (ou *brain-controlled buttons*). Une telle ICM serait semblable à celle utilisée pour la navigation spatiale, mais avec un nombre de commandes accru.

²⁰Dans [Jeon *et al.*, 2011, Nam *et al.*, 2011], les auteurs étudient sur des essais moyennés, l'effet de la durée de l'imagination sur l'ERD et l'ERS ainsi que leur latéralisation. Ils concluent que l'ERS pourrait être utile pour les ICM.

3.2 Réalisation d'une base de données d'enregistrements EEG lors de mouvements brefs

Utilisation hors-ligne d'enregistrements Nous avons expliqué au début de ce chapitre que l'étude hors-ligne de l'imagination continue de mouvements est fortement biaisée par l'absence de feedback. En effet, grâce au feedback reçu, l'utilisateur peut moduler les commandes qu'il envoie. C'est d'ailleurs un des intérêts de ce genre d'ICM : le signal de contrôle peut être modulé en intensité. L'apport des études hors-ligne dans les ICM basées sur des mouvements continus est donc limité, ou du moins difficile.

Ceci n'est pas le cas pour l'ICM basée sur l'imagination de mouvements brefs que nous développons. Bien que l'utilisation en-ligne, avec un feedback, puisse modifier les signaux, la tâche reste de même nature. En effet, le feedback intervenant après l'action, le sujet ne peut pas l'utiliser pour modifier la commande. Par contre il est possible (et l'on espère) que le sujet utilise le feedback pour apprendre à contrôler l'interface et à générer des signaux de contrôle plus clairs. Cependant cet apprentissage est plus lent et devrait être compatible avec des études hors-ligne. Cela ne veut pas dire qu'il n'est pas nécessaire d'effectuer des études en-ligne, mais simplement que les résultats obtenus par des analyses hors-ligne seront plus facilement réutilisables en-ligne.

3.2.1 Le protocole de stimulation

Construction du protocole Le but est de réaliser un protocole de stimulation permettant des études hors-ligne aussi proches que possible d'une utilisation en-ligne et, dans la mesure du possible, compatibles avec une utilisation asynchrone. Il s'agit donc d'enregistrer l'imagination de différents mouvements brefs entrecoupés de périodes de repos.

Idéalement, pour être compatible avec une utilisation asynchrone, il faudrait que le sujet réalise les tâches spontanément. Cependant, le but étant d'obtenir une base de données de mouvements imaginés, il ne serait alors pas possible de savoir de façon certaine quand le sujet effectue la tâche ou quel mouvement il imagine. Un protocole a donc été mis en place pour présenter des consignes au sujet lui demandant d'effectuer certains mouvements. Après plusieurs essais, il nous a paru plus adapté de ne pas annoncer l'arrivée des consignes et de les espacer d'un délai pseudo-aléatoire.

Ce nouveau protocole a deux avantages :

- Il est plus proche d'une utilisation en-ligne d'une ICM asynchrone. Lors d'une utilisation d'une telle ICM, le sujet doit être capable d'envoyer à n'importe quel instant la commande appropriée.
- Il rend les expériences plus stimulantes pour le sujet.

Choix des tâches d'imagination motrices Sur la base des résultats obtenus sur les expériences de mouvements continus et sur celles réalisées dans [Pfurtscheller *et al.*, 2005], nous avons durant les premières sessions d'expérimentations essayé quatre différents mouvements (d'abord avec des mouvements réels puis imaginés) :

- la main gauche,
- la main droite,
- les pieds,
- la langue.



Fig. III.11 – Les 3 mouvements demandés aux 10 sujets : la main gauche, la droite, les deux pieds.

Les résultats pour la langue étant peu exploitables nous avons décidé de ne plus l'utiliser. Nous avons donc conservé 3 tâches (voir figure III.11).

Le protocole du point de vue du sujet L'expérience est découpée en blocs de 30 essais. Chaque bloc commence par une pause de 4.5 secondes, durant laquelle une croix de fixation est présente à l'écran, puis les essais s'enchaînent. Chaque essai commence par l'apparition d'une consigne durant 2 secondes puis est suivi d'une pause comprise entre 2.5 et 9.5 secondes où seule une croix de fixation est présente à l'écran. Le sujet doit imaginer réaliser le mouvement représenté par la consigne durant son affichage (soit durant 2 secondes). L'ordre des consignes est déterminé de façon pseudo-aléatoire pour que chaque bloc comprenne 10 consignes pour chaque tâche.

Durée minimale inter-stimuli Les premières expériences furent réalisées avec une période inter-consignes plus courte : entre 1,25 et 6 secondes. Une nouvelle consigne pouvait donc survenir 1,25 secondes après un mouvement. Alors que l'ERS n'est pas complètement terminé, un nouveau mouvement est imaginé générant normalement un ERD. Il semblerait que cela n'influe pas sur l'ERD associé au deuxième mouvement mais simplement raccourcisse la durée de l'ERS généré par le premier mouvement. Cependant un nombre insuffisant de tels enregistrements furent réalisés pour confirmer cette hypothèse.

Pour le protocole final, nous avons choisi d'augmenter à 2,5 secondes l'écart

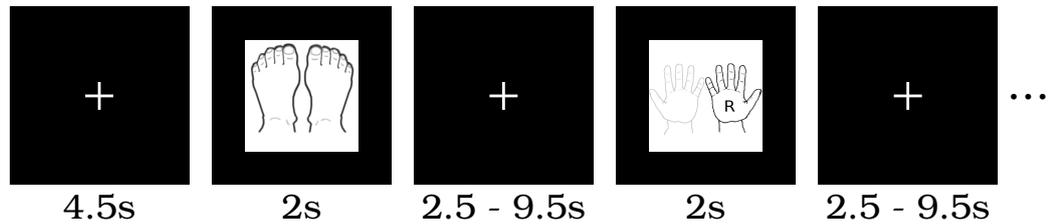


Fig. III.12 – Début d'un bloc du point de vue du sujet. L'enchaînement pseudo-aléatoire des consignes et la variation du temps les séparant obligent le sujet à rester attentif.

minimal entre deux consignes pour éviter l'interaction entre les différentes tâches.

Période de repos L'intervalle situé entre deux consignes permet d'enregistrer l'ERS produit par le mouvement, mais sert aussi pour enregistrer la période de repos (c'est à dire celle où l'utilisateur n'essaie pas d'envoyer de commandes).

	fenêtre tâche	fenêtre post-tâche
Mouvement :	ERD	ERS
Pas de mouvement :		
Avant un mouvement :		ERD
Après un mouvement :	ERS	

Fig. III.13 – Biais favorable à la détection de repos situés près des mouvements. Deux fenêtres peuvent être utilisées pour différencier un mouvement d'un repos : une fenêtre durant l'éventuelle tâche, qui sert à mesurer l'éventuel ERD (baisse de puissance), et une fenêtre postérieure pour l'ERS (augmentation de puissance). Si on analyse un repos précédant un mouvement la fenêtre postérieure à ce repos coïncide avec le mouvement et contient donc un ERD et non une activité normale ou un ERS. Il est alors évident qu'il ne pouvait s'agir d'un mouvement. Le principe est inversé pour un repos suivant un mouvement, mais facilite toujours la détection.

Durant les premières expériences des fenêtres de 2 secondes étaient découpées sur l'ensemble de la période inter-mouvements et utilisées pour reconnaître les périodes de repos. Une majorité des fenêtres utilisées comme période de repos suivaient ou précédaient donc directement un mouvement. Comme le montre la figure III.13 cela introduit un biais.

Le protocole final est développé pour que des périodes séparées des mouvements puissent être utilisées comme période de repos. La période séparant les consignes, bien qu'apparaissant uniforme au sujet, est calculée de trois façons (voir figure III.14) :

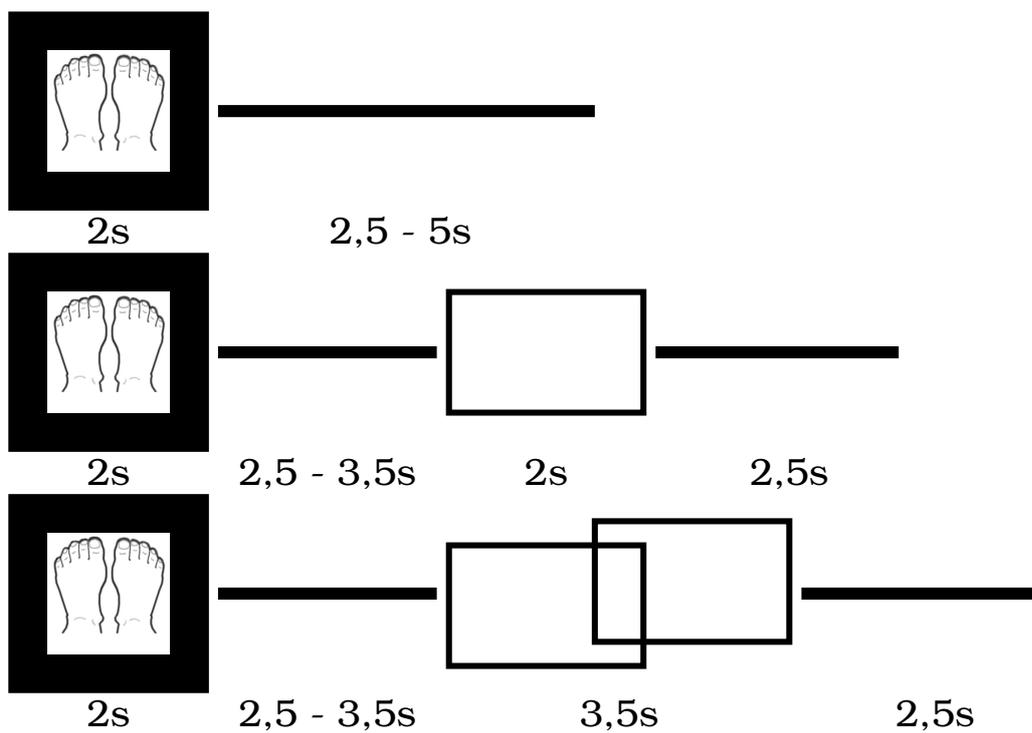


Fig. III.14 – Répartition des fenêtres pour l'apprentissage du repos durant la période inter-consignes. En haut : essai sans période considérée comme de repos, constituant 66% des essais. Au milieu : essai avec une fenêtre de repos, constituant 17% des essais. En haut : essai avec 2 fenêtres de repos, constituant 17% des essais.

- 2 fois sur 3, elle dure entre 2,5 et 5 secondes et aucune période de repos n'est enregistrée.
- 1 fois sur 6, elle dure entre 7 et 8 secondes et une fenêtre de 2 secondes est utilisée comme période de repos.
- 1 fois sur 6, elle dure entre 8.5 et 9.5 secondes et 2 fenêtres de 2 secondes sont utilisées comme périodes de repos.

3.2.2 Détails des enregistrements

Les sujets Après avoir effectué plusieurs expériences pour mettre le protocole au point, nous avons effectué une série d'enregistrements avec 10 sujets : 3 femmes et 7 hommes d'un âge moyen de 32.9 ± 9.4 ans dont un gaucher. Aucun d'entre eux ne souffrait de trouble moteur. Seulement un d'entre eux avait participé précédemment à une expérience d'ICM avec feedback.

Stimulation Le scénario OpenViBE utilisé est le même que pour l'enregistrement de mouvements continus (figure III.4 page 74). Seul le scénario de stimulation a été modifié pour être adapté au nouveau protocole.

Enregistrement du signal Un amplificateur 64 voies ANT est utilisé pour enregistrer le signal à 512Hz. Afin de permettre l'exploration de différentes méthodes de filtrage spatial, 29 électrodes réparties autour du cortex moteur sont enregistrées (figure III.15).

Pour s'assurer que le sujet n'effectue qu'une imagination du mouvement et non une contraction des muscles, trois électrodes bipolaires sont placées sur les avant-bras gauche et droit (muscle brachio-radialis) et le mollet gauche (muscle tibial antérieur) permettant d'enregistrer un électromyogramme (EMG).

Ayant eu quelques problèmes de timing avec OpenViBE, une photodiode est placée sur le coin en bas à gauche de l'écran. Chaque affichage de consigne est couplé à un carré blanc sous la photodiode ce qui permet d'enregistrer l'instant exact où la consigne apparaît à l'écran.

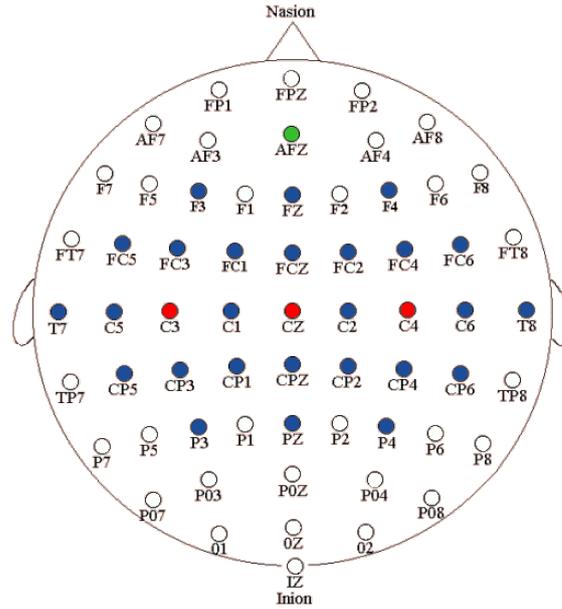
3.3 Analyse temps-fréquence des données

3.3.1 Vision globale simplifiée de l'ERD/ERS

D'après les premières expériences réalisées sur les mouvements brefs et la littérature sur les rythmes sensori-moteurs²¹ le phénomène d'ERD/ERS provoqué par l'imagination d'un mouvement d'un membre intervient dans l'aire corticale associée au membre et est composée de deux phases distinctes :

²¹[Jasper et Andrew, 1938, Pfurtscheller et Aranibar, 1979, Graimann *et al.*, 2002, Pfurtscheller *et al.*, 2005, Pfurtscheller et Solis-Escalante, 2009]

Fig. III.15 – Les électrodes utilisées sont C3, Cz et C4 (en rouge). Pour permettre l'utilisation de filtre spatiaux 26 autres électrodes sont enregistrées (en bleu). L'électrode AFZ (vert) est utilisée comme référence.



- Durant le mouvement :
 - une diminution importante de l'activité dans le μ (ERD),
 - une diminution de l'activité dans le β (ERD).
- Postérieur au mouvement :
 - un retour parfois supérieur à la normale de l'activité dans le μ (très peu décrit dans la littérature),
 - une augmentation forte de l'activité dans le β (ERS, ou rebond β).

La figure III.16 représente les cartes temps-fréquence sur l'électrode C4 (au dessus du cortex sensorimoteur droit) moyennées sur les essais d'imagination de mouvements de la main gauche par deux sujets. L'ERD et l'ERS dans les bandes μ et β y sont particulièrement bien visibles.

T-test sur les cartes temps-fréquence Afin de mieux visualiser les bandes de fréquences pouvant être utiles pour la reconnaissance et la discrimination entre les tâches d'imagination motrice nous avons effectué des t-test de Welch sur les cartes temps-fréquence :

$$T(f, t) = \frac{\overline{P}_1(f, t) - \overline{P}_2(f, t)}{\sqrt{\frac{S_1(f, t)^2}{n_1} + \frac{S_2(f, t)^2}{n_2}}}$$

où $\overline{P}_i(f, t)$ et $S_i(f, t)^2$ sont la moyenne et la variance de la puissance à la bande de fréquences f et l'instant t estimé pour la classe i . Pour éviter une perturbation liée à la présence d'artefacts, les *outliers* de chaque classe sont retirés avant le calcul de la moyenne et de la variance. La valeur signée du t-test est affichée (et non sa valeur absolue) pour permettre de distinguer les augmentations des diminutions de puissance.

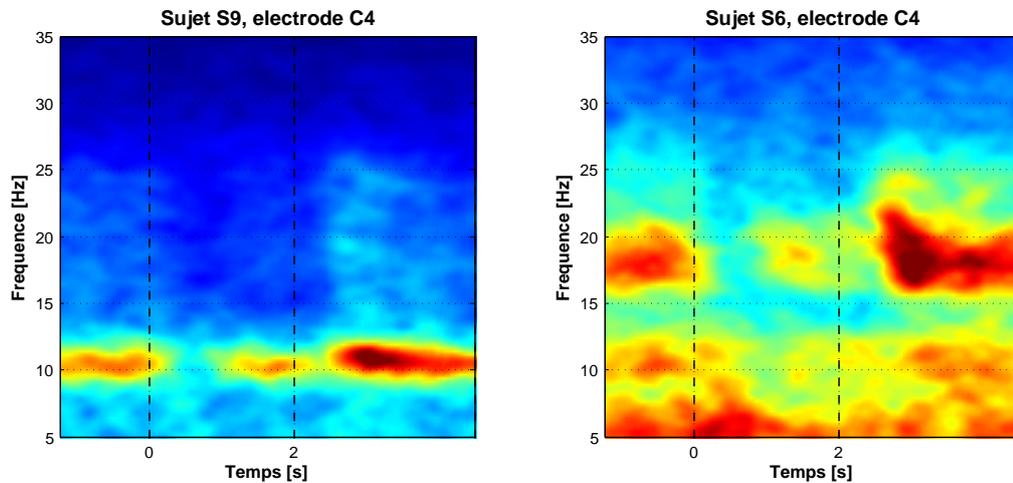


Fig. III.16 – Cartes temps-fréquence moyennes enregistrées sur l'électrode C4 (au dessus du cortex sensorimoteur droit) pour 100 essais d'imagination de mouvements de la main gauche. L'affichage de la consigne de mouvement a lieu de 0 à 2 secondes. À gauche : pour le sujet S9 la bande de fréquence μ est particulièrement visible et l'on y distingue clairement une diminution de puissance (ERD) durant l'imagination du mouvement suivi d'une augmentation (ERS). La bande de fréquences β est moins marquée mais on peut distinguer une différence de puissance entre la période de mouvement et les instants suivants. À droite : pour le sujet S6 l'ERD sur les deux bandes μ et β est visible, par-contre l'ERS n'est visible que sur le β .

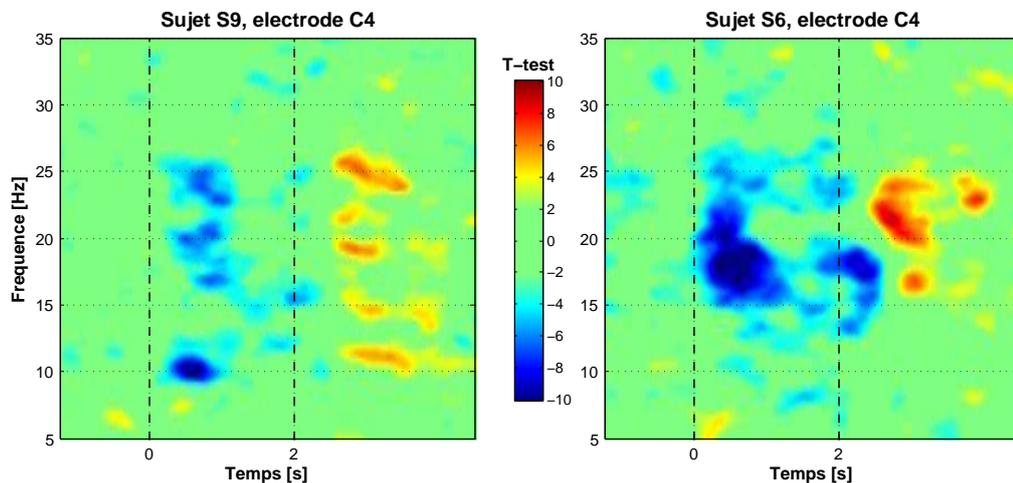


Fig. III.17 – *T*-tests de Welch comparant l'imagination de mouvement de la main gauche au repos. Cette figure montre l'information discriminante comprise dans la figure III.16. À gauche : pour le sujet S9 on constate une baisse puis une hausse de puissance dans quasiment toute la bande fréquences 8-35Hz. À droite : pour le sujet S6 seule l'information comprise dans les bandes de fréquence 15-25Hz est significative.

La figure III.17 représente une carte temps-fréquence de t-tests comparant un mouvement de la main gauche par rapport au repos. Cette carte permet de localiser l'information discriminante contenue dans la figure III.16. Il apparaît que l'ERD durant le mouvement puis l'ERS qui suit peuvent être utilisées pour distinguer un mouvement de la main gauche d'une période de repos en utilisant l'électrode C4.

Il est courant d'utiliser, pour détecter l'imagination de mouvement, un filtrage de 8 à 30Hz englobant les bandes de fréquences μ et β . Un tel filtrage semble particulièrement adapté pour le sujet S9 voire pour le sujet S6. Cependant l'analyse détaillée des cartes temps-fréquence de l'ensemble des 10 sujets nous a montré que le phénomène d'ERD/ERS peut être beaucoup plus complexe et qu'il varie fortement suivant les sujets.

3.3.2 L'ERD/ERS : un phénomène complexe et sujet dépendant

Exemple pour le sujet S8 La figure III.18 montre les cartes temps-fréquence calculées pour le sujet S8 pour les périodes de repos et les trois types de mouvements. L'ERD durant le mouvement est particulièrement marquée sur les électrodes controlatérales durant l'imagination de mouvements des mains et est visible sur les électrodes ipsilatérales et centrales.

La figure III.19 représente les cartes de t-tests pour la détection de chaque tâche (imagination d'un mouvement vs période d'inactivité) ainsi que pour la discrimination entre les tâches. L'analyse des cartes de t-tests, montre que l'ERD est significatif dans toute la bande de fréquences 8-30Hz pour la détection de mouvement des mains sur les électrodes controlatérales et dans la bande 8-25Hz sur les électrodes ipsilatérales et centrales.

Cependant un phénomène nouveau apparaît sur les électrodes C3 et C4 durant l'imagination des pieds : bien qu'un ERD ait lieu sur les bandes de fréquences 8-10Hz et 16-20Hz, un ERS a lieu autour de 14Hz durant le mouvement. Ce phénomène est décrit très brièvement dans [Pfurtscheller et Neuper, 2010], où les auteurs font une distinction entre deux bandes de fréquences dans le μ .

Un filtrage entre 8 et 30Hz amènerait une compensation de l'ERD et de l'ERS intervenant durant l'imagination du mouvement des pieds et donc à une perte d'information. D'une façon plus générale l'analyse des cartes de t-tests pour la distinction entre les diverses tâches fait ressortir des phénomènes plus complexes qui ne pourront pas être utilisés avec un filtrage dans une seule bande de fréquences. Par exemple, toujours pour le sujet S8, durant l'imagination du mouvement des mains un léger ERS autour de 30Hz se produit sur l'électrode Cz alors qu'un ERD se produit durant l'imagination d'un mouvement des pieds. Ainsi la bande 27-30Hz de l'électrode Cz est significative pour différencier un mouvement des pieds de celui des mains, mais une telle information serait perdue avec un filtrage 8-30Hz.

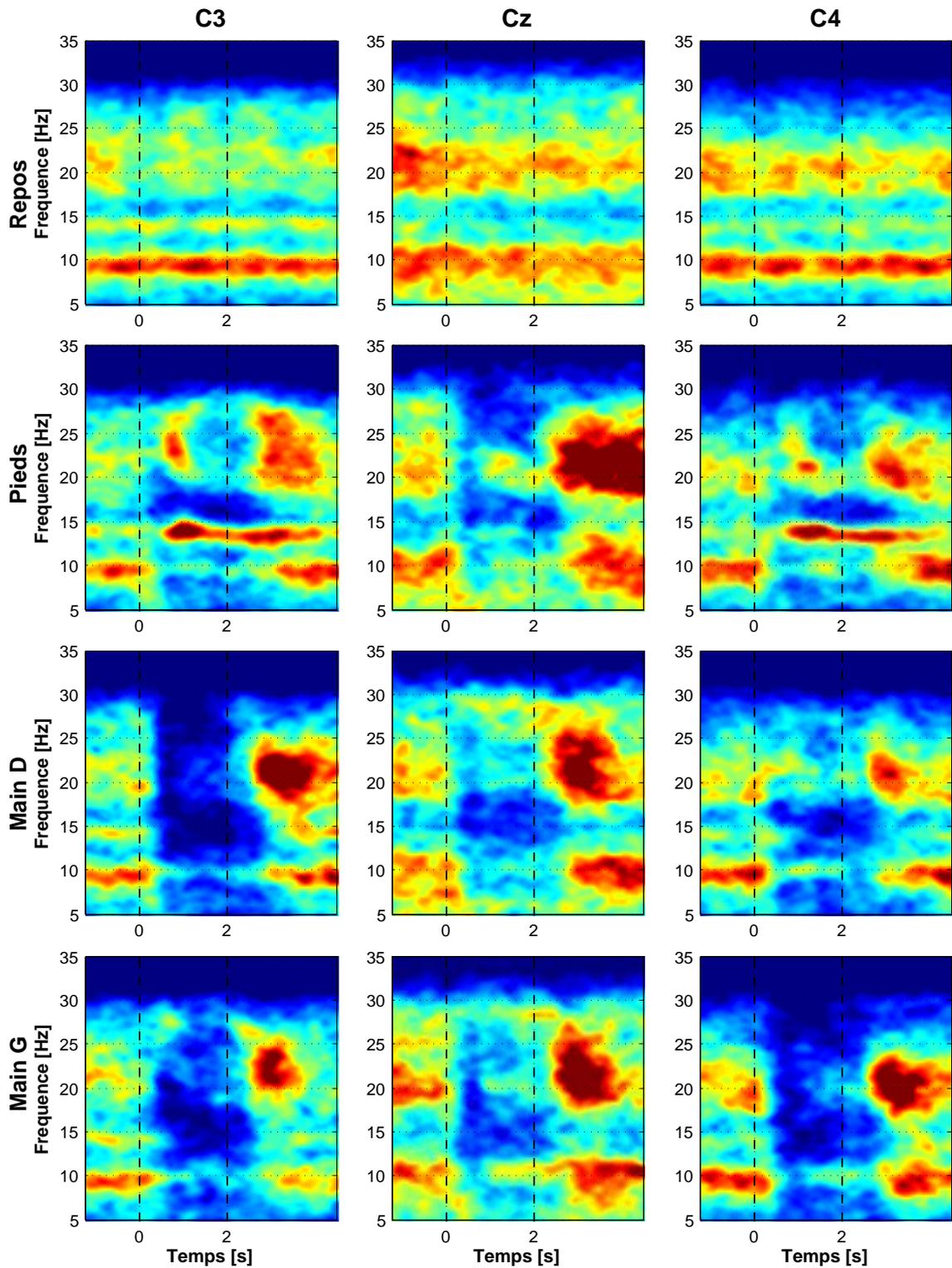
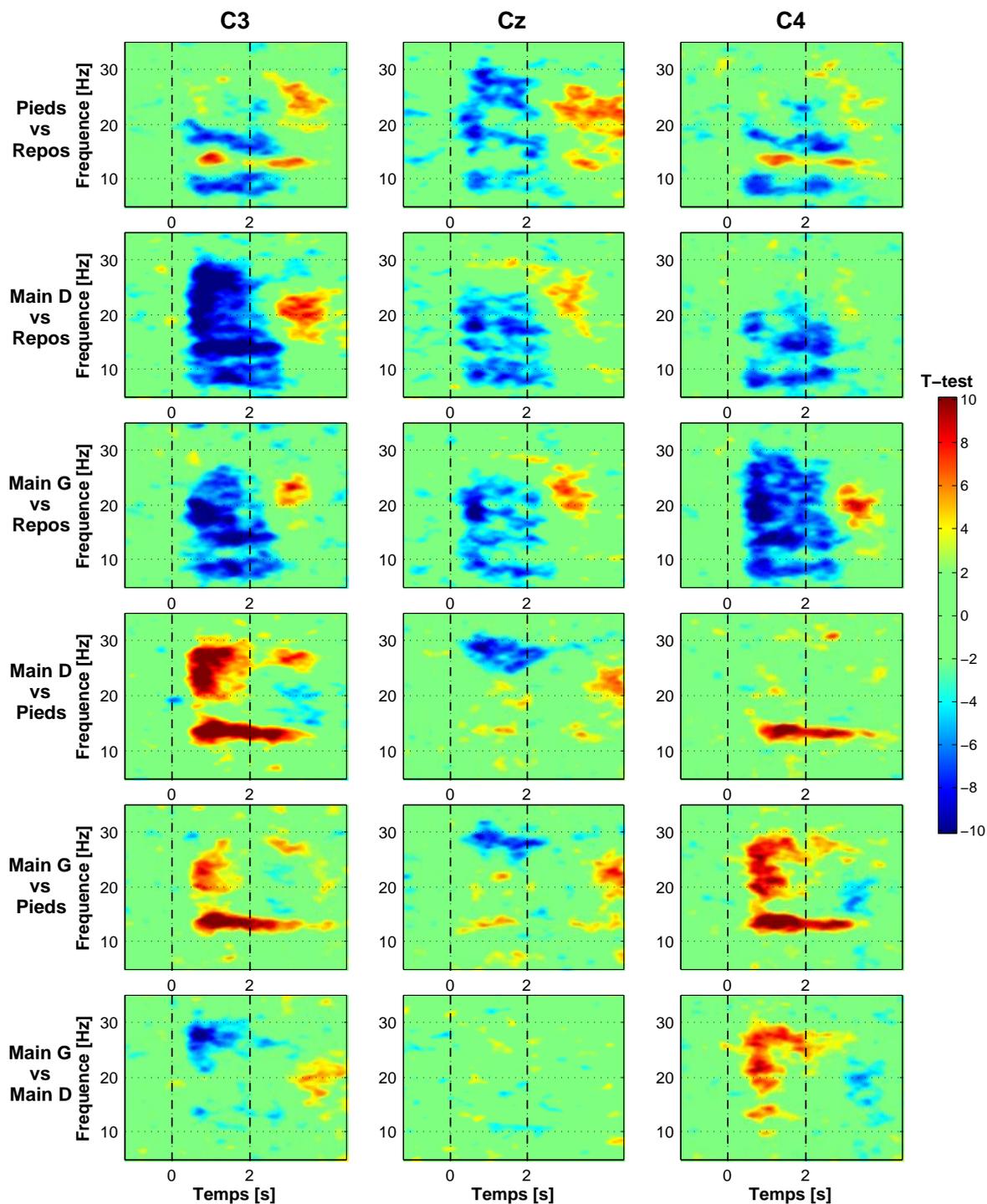


Fig. III.18 – Cartes temps-fréquence pour le sujet S8 lors de l’imagination de mouvements des deux mains et des pieds. L’affichage de la consigne de mouvement a lieu de 0 à 2 secondes. Aucune consigne n’est affichée pour les périodes de repos (voir figure III.12). Les cartes sont calculées sur 100 essais puis moyennées.

Fig. III.19 – Cartes des t -tests pour le sujet S8.

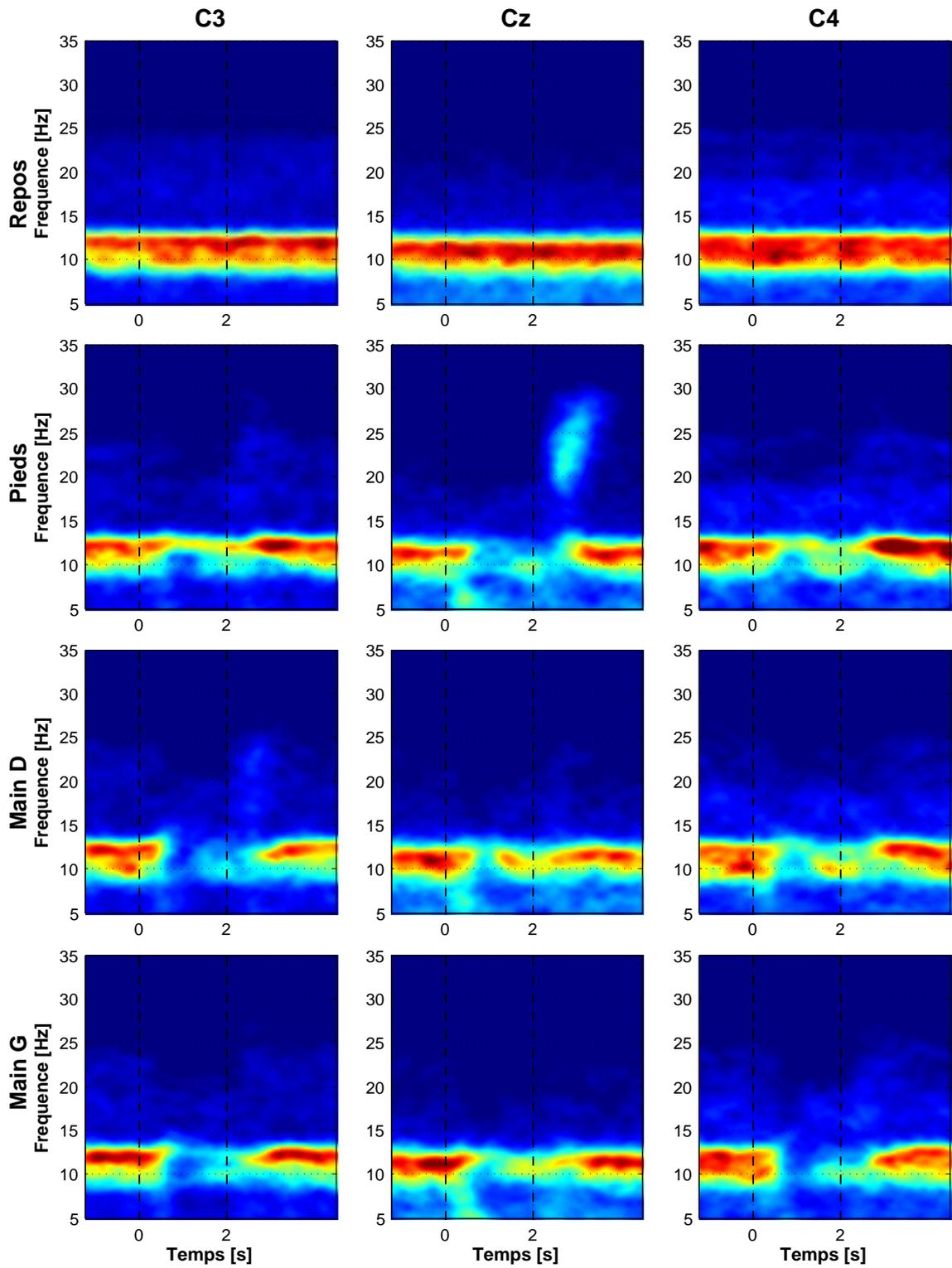


Fig. III.20 – Cartes temps-fréquence pour le sujet S1 lors de l'imagination de mouvements des deux mains et des pieds.

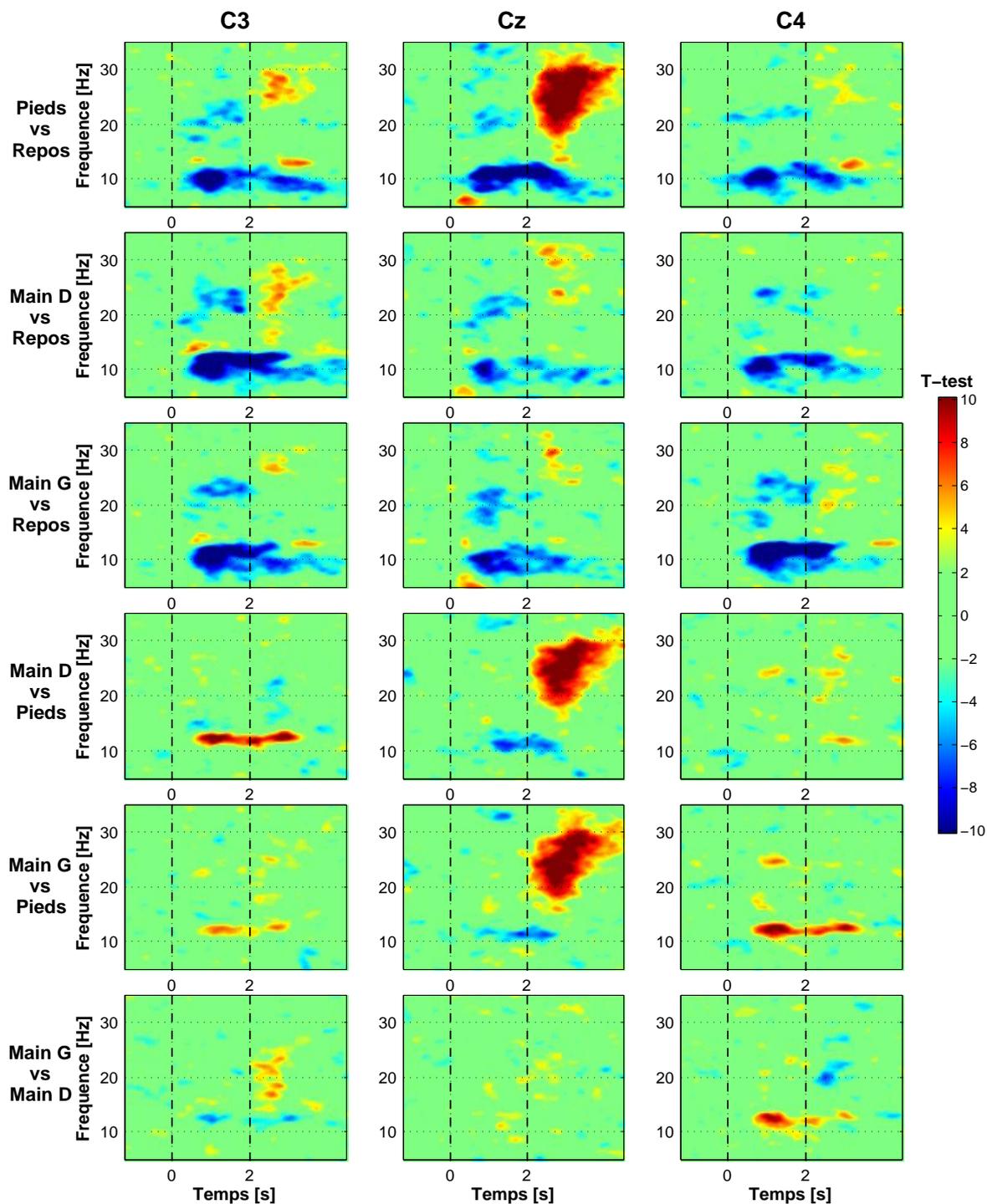


Fig. III.21 – Cartes des t -tests pour le sujet S1.

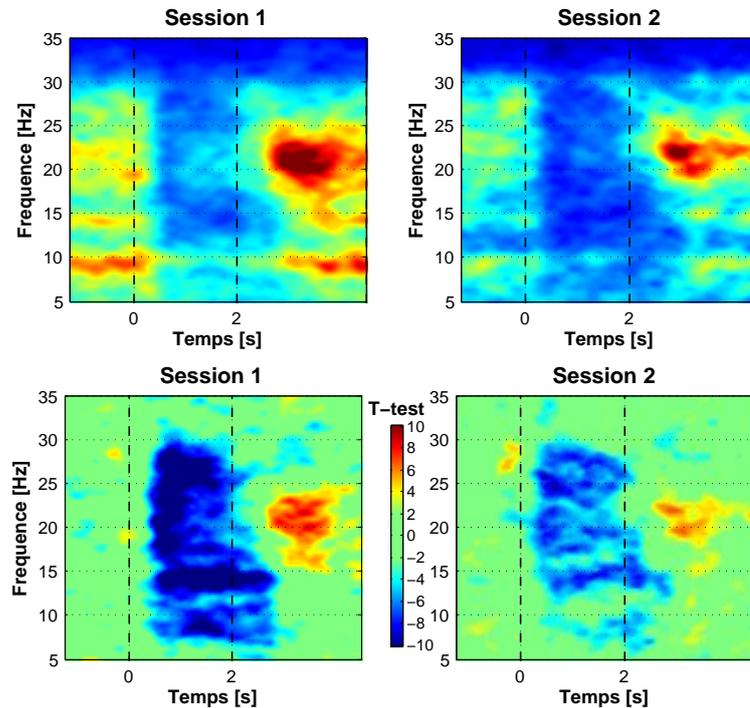


Fig. III.22 – Cartes temps-fréquence (en haut) et de t -test (en bas) enregistrées sur l'électrode C3 (au dessus du cortex sensorimoteur gauche) pour la détection de l'imagination de mouvements de la main droite au cours de deux sessions pour le sujet S8. La puissance moyenne mesurée dans l'ensemble des fréquences, ainsi que la valeur maximale des t -tests, est inférieure pour la deuxième session que pour la première, cependant les patrons restent identiques.

Exemple pour le sujet S1 Les figures III.20 et III.21 représentent les cartes temps-fréquence et de t -tests pour le sujet S1. La bande de fréquence μ est particulièrement bien visible et assez haute (10-14Hz) contrairement à la bande de fréquence β invisible durant le repos. Un rebond β particulièrement marqué intervient après l'imagination des pieds sur l'électrode Cz.

Pour ce sujet, l'ERD dans le μ se poursuit même après la fin de l'imagination du mouvement, alors que se produit déjà un ERS dans le bêta. Ceci a lieu pour tous les mouvements et sur les trois électrodes. L'utilisation d'un filtre moyennant la puissance dans le μ et le β serait une nouvelle fois particulièrement inefficace.

3.3.3 Faible variabilité inter-sessions

Pour 6 des 10 sujets²², une seconde session d'enregistrements fut réalisée pour évaluer le fonctionnement en-ligne de l'ICM (voir section 3.5). Bien que plus d'un

²²Deux sujets n'ont pas été retenus pour l'expérience en-ligne car leurs performances n'étaient pas suffisantes pour contrôler une ICM à 3 commandes et deux sujets n'étaient plus disponibles.

mois se soit écoulé entre les deux séries d'enregistrements, les nouvelles cartes temps-fréquence et de t-tests sont extrêmement semblables à celles réalisées durant le premier enregistrement pour l'ensemble des sujets (voir figure III.22).

3.4 Traitement hors-ligne des données

3.4.1 Chaîne de traitement hors-ligne

L'analyse hors-ligne des données est effectuée avec la boîte à outils *MoffBCI* (qui fait l'objet du chapitre VI). Une chaîne de traitements similaire à celle de l'analyse des mouvements soutenus est utilisée. Cependant, une nouvelle méthode a été ajoutée pour l'extraction des features afin de s'adapter à la diversité du phénomène d'ERD/ERS chez les différents sujets : un spectrogramme est calculé pour chaque essai et chaque atome est utilisé comme feature. Cela génère plusieurs centaines de features par électrode²³.

1. Prétraitement : un filtrage laplacien de taille variable est automatiquement appliqué si celui-ci permet une amélioration des résultats.
2. Extraction des features :
Entre 3 et 20 électrodes situées au dessus du cortex moteur sont utilisées.
Les features sont les valeurs du spectrogramme des puissances de 6 à 30Hz durant l'imagination du mouvement et les 2 secondes suivantes.
3. Sélection des features : un algorithme linéaire récursif basé sur le t-test de Welch est utilisé (voir la section 4.2 du chapitre V page 152).
4. Classification : un SVM avec un noyau linéaire est utilisé.

3.4.2 Détection et discrimination de mouvements

Les taux de détection et de discrimination entre l'imagination des différents mouvements pour l'ensemble des sujets sont donnés dans les tableaux III.3 et III.4.

Pour l'ensemble des sujets au moins l'une des trois tâches est détectable avec plus de 75% de réussite quand les 2 secondes de signal postérieures à l'imagination du mouvement sont utilisées. Ce taux de détection est même supérieur à 85% pour 7 des 10 sujets et supérieur à 90% pour 4. Les tâches permettant les meilleurs résultats varient selon les sujets et il n'y a pas de corrélation significative avec la main directrice des sujets.

Pour la discrimination entre paires de tâches, la moitié des sujets arrivent à un taux de classification supérieur à 95% pour une paire de tâches. Les couples de tâches permettant la meilleure discrimination font toujours intervenir les pieds (mis à part pour le sujet S3 quand seule la fenêtre d'imagination est utilisée).

²³Ces paramètres sont réglés automatiquement. Environ 15 pas de temps et 20 bandes de fréquences sont utilisés pour le calcul du spectrogramme

Sujet	durant imagination			durant + post imagination		
	Pieds	Main D	main G	Pieds	Main D	main G
S1	82%	85%	84%	95%	86%	89%
S2	77%	77%	79%	87%	89%	88%
S3	68%	80%	64%	75%	75%	66%
S4	81%	76%	76%	82%	80%	85%
S5	68%	66%	60%	78%	73%	72%
S6	68%	66%	80%	92%	81%	88%
S7	69%	62%	60%	76%	71%	71%
S8	86%	95%	95%	91%	97%	95%
S9	80%	68%	75%	94%	90%	91%
S10	70%	79%	84%	86%	84%	81%
Moyenne	75%	75%	76%	85%	83%	83%

Tab. III.3 – Taux de classification pour la détection de chaque tâche sans et avec l'utilisation d'une période de 2 secondes suivant le mouvement. Les meilleurs taux de détection sont affichés en gras. Tous les résultats sont significativement différents du hasard ($p > 95\%$).

Sujet	durant imagination			durant + post imagination		
	G vs D	G vs P	D vs P	G vs D	G vs P	D vs P
S1	67%	77%	84%	84%	94%	95%
S2	(51%)	66%	63%	(54%)	76%	69%
S3	74%	68%	71%	70%	66%	72%
S4	(58%)	91%	92%	(57%)	96%	92%
S5	67%	70%	67%	73%	76%	83%
S6	70%	81%	75%	88%	96%	90%
S7	(54%)	71%	68%	(62%)	68%	68%
S8	87%	90%	98%	92%	93%	98%
S9	62%	81%	74%	82%	97%	95%
S10	59%	73%	69%	(52%)	78%	76%
Moyenne	65%	77%	76%	71%	84%	84%

Tab. III.4 – Taux de classification pour la discrimination entre chaque paire de tâche sans et avec l'utilisation d'une période de 2 secondes suivant le mouvement. P : pieds, G : main gauche et D : main droite. Les meilleurs taux de classification sont affichés en gras. Les résultats entre parenthèses ne sont pas significativement différents du hasard ($p = 95\%$).

3.4.3 Gain lié à l'utilisation de la période post-imagination

Sujet	Gain	Intervalle de confiance
S1	10%	[8 ; 13]
S2	8%	[6 ; 11]
S3	0%	[-3 ; 3]
S4	3%	[0 ; 5]
S5	9%	[6 ; 12]
S6	16%	[13 ; 19]
S7	5%	[2 ; 9]
S8	2%	[1 ; 4]
S9	18%	[15 ; 21]
S10	4%	[0 ; 8]
Moyenne	7.6%	[6.6 ; 8.6]

Tab. III.5 – Gain moyen, pour la classification binaire entre toutes les paires de tâches et le repos, obtenu grâce à l'utilisation des 2 secondes de signal suivant l'imagination des mouvements. (Les intervalles de confiance sont donnés à $p = 95\%$ et sont évalués grâce aux résultats sur les différents ensembles de la validation croisée). Les gains en gras sont significatifs.

L'utilisation d'une fenêtre de 2 secondes suivant la période d'imagination des mouvements permet un gain de 8.2% ([6.9;9.5] à $p = 95\%$) pour la détection des mouvements et de 7% ([5.5;8.5] à $p = 95\%$) pour la différenciation entre les paires de tâches en moyenne sur l'ensemble des sujets.

Le détail des gains pour chaque sujet est donné dans le tableau III.5.

3.5 Détection et discrimination en-ligne de deux tâches

L'utilisation de l'imagination de mouvements brefs pour les ICM que nous avons entreprise durant cette thèse est poursuivie par Eoin Thomas, post-doctorant de l'équipe. Une nouvelle session d'enregistrements a été effectuée pour 6 des 10 sujets avec un traitement en-ligne des données et la production d'un feedback. Les deux meilleures tâches sont utilisées pour chaque sujet. Il s'agit donc de détecter l'imagination des mouvements puis de distinguer entre les deux types de mouvements possibles. Afin de permettre une évaluation des performances, l'affichage de consignes (ou *cues*) indique au sujet quand effectuer l'imagination.

Les résultats sont résumés dans le tableau III.6. Un article détaillant les résultats est en cours de soumission. Ceci confirme qu'il est possible de commander plusieurs boutons par l'imagination de mouvements brefs.

Sujet	Hors-ligne	En-ligne	Tâches
S1	82.2	76.6	Main D, Pieds
S4	71.2	68.5	Main G, Pieds
S5	67.3	56	Main D, Main G
S6	85	71.6	Main D, Pieds
S8	88.6	73.1	Main D, Main G
S9	86.6	75.9	Main D, Pieds

Tab. III.6 – Performances pour les 6 sujets qui ont participé à l’expérience en-ligne. Les performances hors-ligne sont calculées par validation croisée sur les données acquises lors de la première session d’enregistrement. Cette session d’enregistrement permet de sélectionner les deux meilleures tâches parmi trois et d’effectuer l’apprentissage avant l’expérience en-ligne. Bien que deux mois se soient écoulés avant la session en-ligne, aucun nouvel apprentissage n’a été effectué. (Il s’agit de classification à trois classes, 2 tâches + repos, le taux de classification aléatoire est donc de 33%.)

3.6 Conclusion et Perspectives

3.6.1 Analyse des résultats

Détection et discrimination de mouvements brefs imaginés Le travail effectué montre qu’il est non seulement possible d’utiliser la période suivant l’imagination motrice pour détecter des mouvements imaginés des pieds et des mains mais que cette information peut aussi être utilisée pour distinguer les tâches entre elles. Les taux de classification obtenus, que ce soit pour la détection ou la discrimination, sont particulièrement élevés, ce qui laisse présager qu’il sera possible d’utiliser l’imagination motrice de mouvements brefs pour contrôler 2 voire 3 boutons par la pensée de façon asynchrone.

Ces résultats montrent qu’il est aussi possible d’augmenter grandement les performances de l’ICM de navigation spatiale décrite dans la section 2 en utilisant l’ensemble du spectre de fréquences durant l’imagination d’un mouvement et la période suivant celle-ci, plutôt que de n’utiliser que le rebond bêta.

ICM fonctionnant sur l’ensemble des sujets testés En moyenne entre 15 et 30% des sujets n’arrivent pas à contrôler une ICM basée sur l’imagination motrice [Dickhaus *et al.*, 2009, Vidaurre et Blankertz, 2010]. On parle généralement de *BCI Illiteracy* qu’on peut traduire par illettrisme des ICM. Sur l’ensemble des 10 sujets, tous arrivent pour au moins une tâche à un niveau de détection suffisant pour permettre la création d’un bouton contrôlé par la pensée. Le nombre de sujets n’est pas suffisant pour affirmer que ce type d’ICM peut fonctionner sur tous les sujets. Cependant, il est possible que l’utilisation de la période post-imagination puisse permettre le contrôle d’une ICM à des sujets n’arrivant normalement pas à

utiliser l'imagination motrice.

Une autre explication peut venir du fait que nous utilisons l'ensemble du spectrogramme pour extraire le signal de contrôle, ce qui permet d'obtenir des features parfaitement adaptés pour tous les sujets. Or d'après [Vidaurre et Blankertz, 2010] l'adaptation de l'ICM au sujet peut être une solution au problème "d'illettrisme des ICM".

Variabilité importante inter-sujets Nous avons effectué durant ce travail de thèse des enregistrements avec une vingtaine de sujets différents. Ce n'est qu'en étudiant en détail les cartes temps-fréquence de l'ensemble des sujets que nous nous sommes aperçus que, pour certains d'entre eux, les phénomènes observés pouvaient être particulièrement différents de ce qui est décrit dans la littérature. Suite à cette constatation nous avons utilisé une méthode d'extraction de features limitant les *a priori* et permettant une meilleure adaptation à chaque sujet. Ceci a permis un gain particulièrement important pour certains sujets.

La diversité inter-sujet étant telle, il nous paraît nécessaire, pour qu'une ICM puisse fonctionner de façon optimale, qu'elle s'adapte de façon automatique aux spécificités de chaque utilisateur. Cette approche est en harmonie avec l'ANR Co-Adapt à laquelle participe partiellement ce travail de thèse. En effet cette ANR a pour but d'étudier la co-adaptation entre l'utilisateur et la machine dans les ICM.

3.6.2 Vers une ICM totalement asynchrone

Bien que les travaux décrits dans cette section aient pour but la réalisation d'ICM asynchrone, nous avons dû utiliser des protocoles synchrones pour faire l'acquisition des données.

L'étape suivante consiste à simuler le fonctionnement asynchrone de l'ICM sur la base de données déjà enregistrée. Pour l'instant les algorithmes ont été testés sur des fenêtres correspondant exactement à l'imagination d'un mouvement ou au milieu de repos. Même si détecter une période repos précédant ou suivant l'imagination d'un mouvement est plus aisé (voir III.13) il est nécessaire de vérifier le comportement des algorithmes sur des fenêtres de signal glissantes. Il est alors probable que des algorithmes de classification utilisant l'évolution temporelle du signal (à l'aide de chaînes de Markov cachées par exemple [Doležal et al., 2011]) soient plus performants.

La dernière étape consiste à utiliser en-ligne l'ICM de façon totalement asynchrone (ce qui a été fait avec l'ICM de navigation spatiale). Les commandes que désire envoyer l'utilisateur sont alors inconnues et il n'est alors plus possible d'effectuer d'apprentissage sur les données ainsi acquises. Nous avons envisagé deux solutions pour poursuivre l'apprentissage durant une utilisation asynchrone et permettre ainsi une adaptation continue de l'ICM.

Utilisation du signal suivant l'imagination pour "étiqueter" les données Pour certains des utilisateurs, les performances en n'utilisant que la période de l'imagination du mouvement (et donc que l'ERD sans l'ERS) sont suffisamment élevées pour permettre le contrôle de l'ICM. D'autre part nous espérons qu'avec l'expérience les utilisateurs arrivent à mieux contrôler l'interface. Ainsi, il est possible qu'après un temps variable, les utilisateurs parviennent à contrôler l'ICM en n'exploitant que la période de l'imagination. Il serait alors possible d'utiliser l'information supplémentaire présente dans les secondes suivant l'imagination, pour effectuer une nouvelle classification plus précise. Cette nouvelle classification (qui pourrait aussi être couplée à la détection de potentiel d'erreurs) peut être utilisée pour :

- Corriger une éventuelle erreur.
- Étiqueter les données pour effectuer un nouvel apprentissage. En d'autres termes, il s'agit d'utiliser l'information suivant l'imagination du mouvement pour détecter, lors d'un protocole asynchrone, quand et quelles commandes le sujet désirait envoyer. Cette connaissance serait utilisée pour effectuer un nouvel apprentissage des algorithmes utilisant uniquement sur la période d'imagination. Cette idée est décrite dans [Fruitet *et al.*, 2011].

Il faudra cependant s'assurer que l'envoi de la commande dès la fin de l'imagination ainsi que l'analyse du feedback, n'interfère pas avec l'ERS.

Apprentissage durant l'utilisation asynchrone d'un jeu Une autre solution pour continuer à faire de l'apprentissage durant l'utilisation d'une ICM asynchrone, est d'utiliser un protocole durant lequel on peut savoir, à certains instants, quel type de commandes le sujet doit envoyer. C'est dans ce but que nous avons développé un jeu utilisant OpenViBE inspiré de *Space Invaders* (voir figure III.23). L'objectif est de laisser suffisamment de liberté à l'utilisateur pour que le jeu soit stimulant, tout en permettant d'étiqueter le plus de données possibles. Si l'on suppose que l'utilisateur joue pour gagner, les commandes qu'il doit produire au cours du temps peuvent être en grande partie déterminées. Une telle approche pourrait être substituée au protocole d'apprentissage conventionnel dès que l'utilisateur atteint un niveau de contrôle minimal de l'ICM.

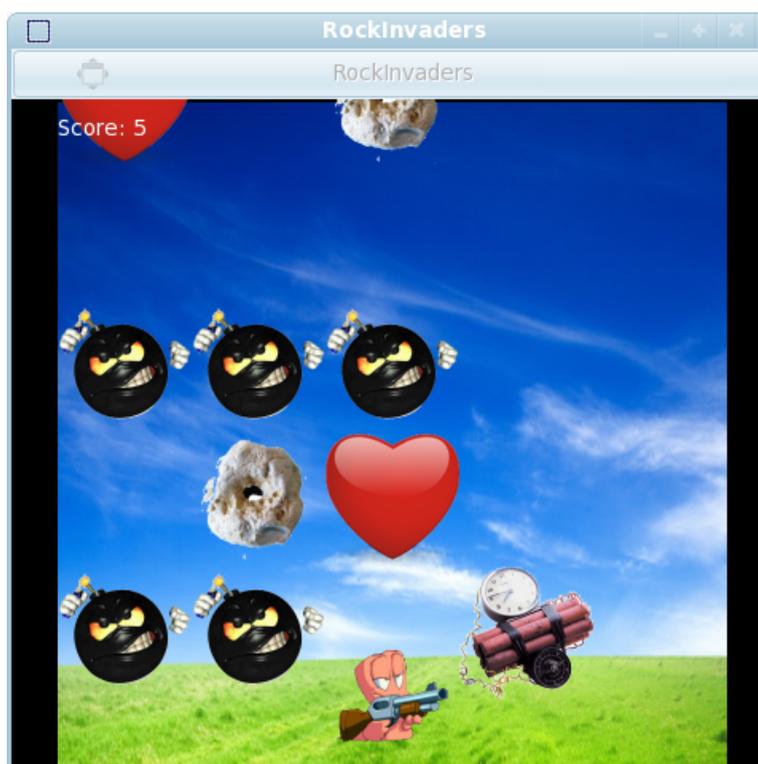


Fig. III.23 – Le jeu “RockInvaders”. Le joueur déplace le personnage de droite à gauche alors que des objets tombent du ciel. Il doit éviter les rochers et bombes tout en attrapant des cœurs. Bien que le sujet soit totalement libre, on peut souvent déterminer les actions qu’il devrait réaliser pour gagner et ainsi étiqueter une partie des données. Dans l’exemple représenté, le joueur ne doit effectuer aucune action, puis se déplacer vers la droite après avoir attrapé le cœur.

4 CONCLUSION

Nous avons étudié dans ce chapitre trois approches différentes pour réaliser des ICM basées sur l'imagination motrice.

La première, la plus commune, utilise l'imagination de mouvements soutenus pendant plusieurs secondes. Nous avons conçu une chaîne de traitement permettant de distinguer différents mouvements basés sur des fenêtres d'enregistrement ne constituant qu'une fraction de la période de production (ou d'imagination) du mouvement. Ceci nous a permis d'étudier l'augmentation du taux de classification en fonction de la durée des fenêtres utilisées, ainsi que l'impact sur le taux de transfert d'information.

La seconde approche que nous avons étudiée est basée uniquement sur le rebond β , produit par l'imagination ou la réalisation de mouvements des pieds ou des mains. En partant d'un transducteur détectant les rebonds β , nous avons conçu une ICM permettant de se déplacer dans un environnement virtuel en 3 dimensions. L'originalité de cette réalisation provient de la conversion d'un signal de contrôle rudimentaire (détection d'un rebond β) en une commande complexe (tâche de navigation spatiale).

Poussés par les résultats encourageants de l'ICM de navigation spatiale, nous avons réalisé un transducteur plus complexe utilisant non-seulement le rebond β mais aussi toutes les informations présentes dans les variations des rythmes sensori-moteurs (ERD et ERS) durant la période d'imagination et la période lui succédant. Nous avons montré qu'il est possible de distinguer de façon fiable l'imagination de plusieurs mouvements brefs de périodes repos, mais aussi les uns des autres. Ceci permet de réaliser ce que nous avons appelé "boutons" contrôlés par la pensée.

Sélection automatique de tâches pour le contrôle d'une ICM

NOUS avons présenté dans le chapitre précédant une nouvelle ICM de type bouton contrôlé par la pensée. Cette ICM est basée sur l'imagination d'un ou plusieurs mouvements pour activer chaque bouton. Afin de garantir un fonctionnement optimal, il est important que les tâches utilisées soient adaptées à chaque sujet. Nous avons, dans ce but, développé un algorithme permettant d'explorer un grand nombre de tâches motrices et de sélectionner automatiquement la plus appropriée¹.

¹ Le travail présenté dans ce chapitre a été réalisé en collaboration avec Alexandra Carpentier dans le cadre de l'ANR Co-Adapt. Elle effectue sa thèse sur la théorie des bandits dans l'équipe SequeL, à l'INRIA de Lille, sous la direction de Rémi Munos.

Sommaire

1	La sélection des tâches contrôlant les ICM	117
1.1	Pourquoi faire de la sélection de tâches?	117
1.2	Un problème exigeant des compromis	117
2	Initiation à la théorie des bandits	118
2.1	La problématique du bandit manchot	118
2.1.1	<i>Vision informelle du problème</i>	118
2.1.2	<i>Formalisation du problème</i>	119
2.2	Le compromis exploration/exploitation	119
2.2.1	<i>Stratégie naïve</i>	119
2.2.2	<i>Exploration et exploitation en continu</i>	119
3	Sélection automatique d'une tâche optimale	120
3.1	Modélisation de notre problème	120
3.2	L'algorithme de sélection de tâches	120
3.2.1	<i>L'algorithme UCB-classif</i>	121
3.2.2	<i>Principe de fonctionnement de l'algorithme UCB-classif</i>	121
3.2.3	<i>Quelques résultats mathématiques sur l'algorithme UCB-classif</i>	122
3.3	Évaluation de performances sur un nombre faible de réalisations	124
3.3.1	<i>Utilisation d'un ensemble de features réduit</i>	125
3.3.2	<i>Essai d'utilisation de mesures de distance entre classes</i>	125
3.3.3	<i>Amélioration de la précision de l'évaluation sur les ensembles de données réduits</i>	125
4	Analyse hors-ligne de l'algorithme UCB-classif	127
4.1	Objectifs et méthode pour l'analyse hors-ligne	127
4.2	Résultats hors-ligne	128
4.2.1	<i>Amélioration du taux de sélection d'une bonne tâche</i>	128
4.2.2	<i>Utilisation optimale du budget</i>	130
5	Sélection d'une tâche optimale en-ligne	131
5.1	Scénario OpenViBE pour la sélection de tâches en-ligne	131
5.1.1	<i>Objectif et principe de fonctionnement</i>	131
5.1.2	<i>Détails sur le scénario OpenViBE</i>	132
5.2	Résultats	133
6	Conclusion et perspectives	135

1 LA SÉLECTION DES TÂCHES CONTRÔLANT LES ICM, UN POINT SOUVENT NÉGLIGÉ

1.1 Pourquoi faire de la sélection de tâches ?

Comme on peut s'en apercevoir à la lecture du chapitre précédent (*Imagination motrice et rythmes sensori-moteurs*), la performance d'un utilisateur se servant d'une ICM basée sur l'imagination motrice varie en fonction de la tâche utilisée. De la même façon qu'il existe des sujets plus ou moins doués pour contrôler une ICM [Dickhaus *et al.*, 2009, Vidaurre et Blankertz, 2010], certaines tâches d'imagination motrices ou tâches cognitives permettent des meilleurs niveaux de performances que d'autres [Le Pavec et Clerc, 2008, Besserve *et al.*, 2011].

Pour l'ensemble des sujets dont nous avons fait passer les expériences, les mouvements imaginés de la langue étaient ceux permettant la moins bonne discrimination. Cependant pour certains sujets les meilleures performances ont été obtenues avec la main droite, alors que pour d'autres c'était avec les pieds. Il existe donc des tâches permettant d'obtenir de meilleures performances que d'autres, mais malheureusement ces "bonnes" tâches varient suivant les sujets.

D'après [Dobrea et Dobrea, 2009, Dobrea *et al.*, 2010] l'utilisation de tâches adaptées à chaque sujet pourrait permettre d'améliorer grandement les performances des ICM. Pourtant, à notre connaissance aucune méthode permettant une sélection de tâche efficace n'a été étudiée à ce jour (un état de l'art sur la sélection de tâche est effectué dans la section 2.4 du chapitre II). Ce chapitre est consacré au développement d'une méthode automatique de sélection d'une tâche optimale adaptée à chaque sujet pour l'utilisation d'un bouton commandé par la pensée.

1.2 Un problème exigeant des compromis

Une méthode de sélection de tâche idéale permettrait :

- d'explorer une grande variété de tâches ;
- de sélectionner la, ou les, tâches statistiquement et significativement meilleures que les autres ;
- de ne pas perdre trop de temps dans l'étape de calibration.

Malheureusement, évaluer les performances d'une tâche nécessite de nombreuses répétitions et ce d'autant plus que l'évaluation doit être précise. Il paraît donc difficile de pouvoir en même temps explorer un grand nombre de tâches tout en garantissant une évaluation fiable des performances de chaque tâche.

L'idée pour effectuer une sélection de tâche efficace est de partir d'un grand nombre de tâches et d'éliminer le plus rapidement possible les tâches sub-optimales afin de se concentrer sur les tâches ayant le meilleur potentiel. Pour réaliser un tel

algorithmes, nous allons utiliser la théorie des bandits, qui est brièvement présenté dans la section suivante.

2 INITIATION À LA THÉORIE DES BANDITS



Fig. IV.1 – Exemple de bandit manchot. Le joueur actionne le bras et reçoit une récompense pécuniaire suivant une loi de probabilité propre à la machine.

Le but de cette section est de donner une intuition de ce qu'est la théorie des bandits et dans la mesure du possible de donner les bases nécessaires pour comprendre comment fonctionne notre algorithme de sélection de tâches. Une présentation plus approfondie de la théorie des bandits est effectuée dans [Robbins, 1952, Auer *et al.*, 2002].

2.1 La problématique du bandit manchot

2.1.1 Vision informelle du problème

La théorie des bandits doit son origine aux machines à sous appelées “bandits manchots” [Morris et de Groot, 1981] (voir figure IV.1).

La problématique originelle peut être exposée de la façon suivante : un joueur se trouve face à plusieurs machines à sous de type bandit manchot et souhaiterait évidemment remporter le plus d'argent possible (ou plutôt en perdre le moins²). Il doit pour cela actionner les différents bras qui s'offrent à lui (c'est le nom des leviers des bandits manchot). Chaque bras lui fournit aléatoirement des récompenses plus ou moins importantes suivant une loi de probabilité qui leur est propre. Les lois de probabilité régissant chaque bras sont a priori différentes et inconnues du joueur. Le but est donc de trouver quel bras permet d'obtenir les meilleures récompenses et de tirer celui-ci un maximum de fois.

²Attention l'addiction au jeu est une réalité, pour plus d'information : <http://aide-info-jeu.fr>

2.1.2 Formalisation du problème

Soit K le nombre de bras. À chaque temps $t \in \mathbb{N}$ on choisit un bras $k_t \in \{1, \dots, K\}$ qui sera actionné et on reçoit une récompense r_t tirée indépendamment suivant la distribution du bras k_t .

Le but est de trouver une stratégie permettant de maximiser l'espérance des gains, soit :

$$\mathbb{E} \left(\sum_{t \in \mathbb{N}} r_t \right)$$

2.2 Le compromis exploration/exploitation

2.2.1 Stratégie naïve

La stratégie qui vient à l'esprit quand on prend connaissance du problème des bandits consisterait à, dans un premier temps, trouver le bras qui rapporte le plus (phase d'exploration), puis ne tirer que celui-ci, afin de maximiser les gains (phase d'exploitation).

Malheureusement, comme les récompenses délivrées par les bras sont stochastiques, évaluer la "performance" d'un bras est coûteux et même impossible à déterminer avec certitude. En d'autres termes, la phase d'exploration (c'est à dire celle qui vise à déterminer quels sont les bras les plus rentables) est potentiellement infinie.

Si on se contente d'une évaluation approximative des performances des bras, on risque durant la phase d'exploitation (c'est à dire celle où l'on tire que le bras ayant la plus forte espérance de gain) de ne pas avoir sélectionné le meilleur bras et donc de ne pas maximiser les gains potentiels.

2.2.2 Exploration et exploitation en continu

La théorie des bandits donne un résultat qui peut paraître surprenant vis à vis du compromis exploration/exploitation : les phases d'exploration et d'exploitation doivent se faire en parallèle et il ne faut jamais s'arrêter d'explorer. En d'autres termes il faut continuer à régulièrement essayer des bras sub-optimaux pour vérifier qu'ils sont bien moins performants que le bras ayant été évalué comme optimal.

3 SÉLECTION AUTOMATIQUE D'UNE TÂCHE OPTIMALE BASÉE SUR LA THÉORIE DES BANDITS

Nous avons développé, en collaboration avec Alexandra Carpentier, un algorithme permettant de sélectionner au cours d'une expérience EEG la tâche la plus appropriée pour réaliser un bouton commandé par la pensée. Cette partie est consacrée à l'explication du fonctionnement théorique de l'algorithme.

3.1 Modélisation de notre problème

Soit K le nombre de tâches parmi lesquelles on veut sélectionner celle que l'on détecte le mieux par rapport au repos. Pour cela, on va demander à l'utilisateur d'effectuer plusieurs fois chacune des tâches afin d'évaluer leur performance, définie par le taux de détection vis à vis du repos. Notons N le nombre total de réalisations de tâches que l'on s'accorde pour sélectionner la tâche la plus appropriée.

Une stratégie de sélection $(k_i)_{1 \leq i \leq N}$ consiste à définir au cours du temps quelles tâches on va demander à l'utilisateur d'effectuer, soit, pour chaque $t \in \{1, \dots, N\}$ quelle est la tâche $k_t \in \{1, \dots, K\}$ que le sujet doit réaliser.

Le but est de trouver une stratégie de sélection qui permette de déterminer le plus rapidement possible la tâche donnant les meilleures performances. D'autre part, les données enregistrées durant la sélection de tâche sont aussi réutilisées pour l'apprentissage de la classification avant l'utilisation de l'ICM. On veut donc aussi maximiser le nombre de fois où la tâche finalement sélectionnée est présentée. Ce problème est donc similaire au problème du "bandit stochastique" [Robbins, 1952, Auer et al., 2002].

Modélisation de la sélection de tâche

On modélise les K différentes tâches comme K bras d'un bandit et on définit la récompense obtenue pour le tirage du bras k comme le taux de détection de la tâche associée.

Dans le modèle du "bandit stochastique", tirer un bras donne une récompense stochastique que l'on utilise pour estimer la distribution associée à ce bras. Dans notre cas, quand on demande au sujet d'effectuer une tâche, cela fournit un nouvel échantillon de données que l'on utilise pour évaluer plus finement le taux de détection de cette tâche.

3.2 L'algorithme de sélection de tâches

L'algorithme développé pour la sélection de tâche est une variante de l'algorithme "Upper Confidence Bound" [Auer et al., 2002]. Le principe de cet algorithme est de construire une borne de confiance supérieure à la récompense de chaque bras et de sélectionner à chaque étape le bras ayant la borne la plus haute.

Nous allons dans un premier temps décrire l'algorithme *UCB-classif*, puis expliquer son principe de fonctionnement et enfin donner une preuve de son efficacité.

3.2.1 L'algorithme *UCB-classif*

L'algorithme *UCB-classif*

Paramètres : a, N, q

Effectuer chaque tâche q fois (soit $T_{k,qK} = q$).

for $t = qK + 1, \dots, N$ **do**

Évaluation par validation croisée à q ensembles le taux de détection $\hat{r}_{k,t}$ de chaque tâche.

Calcul de la borne supérieure : $B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a \log N}{T_{k,t-1}}}$ pour chaque action $1 \leq k \leq K$.

Sélection de la tâche à exécuter : $k_t = \arg \max_{k \in \{1, \dots, K\}} B_{k,t}$.

Mise à jour : $T : T_{k_t,t} = T_{k_t,t-1} + 1$ et $\forall k \neq k_t, T_{k,t} = T_{k,t-1}$

end for

Tab. IV.1 – Pseudo-code de l'algorithme *UCB-classif*.

Notons $B_{k,t}$ la borne sur la récompense du bras k à l'instant t . Dans le cas du "bandit stochastique", cette borne est définie par la moyenne empirique des récompenses obtenues pour le bras k à laquelle s'ajoute un intervalle de confiance dépendant du nombre de tirages $T_{k,t-1}$ du bras k qui ont été effectués au temps $t - 1$:

$$B_{k,t} = \hat{r}_{k,t-1} + \sqrt{\frac{a \log N}{T_{k,t-1}}}, \quad (\text{IV.1})$$

où $a > 0$ est une constante.

Dans notre cas, $\hat{r}_{k,t-1}$ est l'estimateur du taux de détection de la tâche k obtenue sur les $T_{k,t-1}$ essais où l'utilisateur a déjà effectué la tâche. Ce taux de détection est estimé par validation croisée (voir section 3.3).

Soit r_k^* le taux de détection théorique du classifieur linéaire optimal (c'est-à-dire le taux de détection qui serait obtenu avec un ensemble d'apprentissage infini). La section suivante montre que $B_{k,t}$ est une borne supérieure de r_k^* avec haute probabilité. Plus précisément, la probabilité $p(B_{k,t} < r_k^*)$ décroît de façon polynomiale vers 0 en fonction de N .

3.2.2 Principe de fonctionnement de l'algorithme *UCB-classif*

La figure IV.2 illustre le fonctionnement de l'algorithme *UCB-classif*.

Deux facteurs entrent en jeu pour la sélection de la prochaine tâche k_t que le sujet devra exécuter :

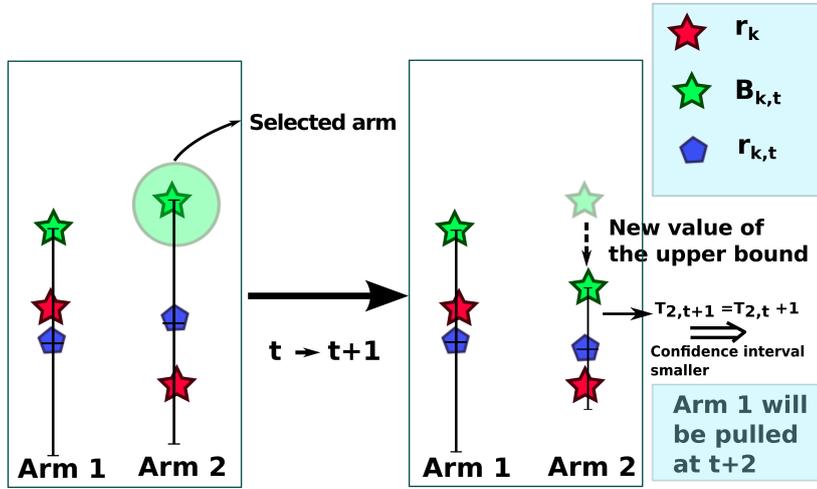


Fig. IV.2 – Fonctionnement de l’algorithme UCB pour un bandit à 2 bras : impact de la sélection du bras 2 au temps t . Même si le bras 1 est plus performant ($r_1^* > r_2^*$, représenté par les étoiles rouges), au temps t , $B_{1,t} < B_{2,t}$. Le bras 2 est donc sélectionné. Cela affine l’estimation de r_2^* : $\hat{r}_{2,t+1}$ et réduit l’intervalle de confiance. Au temps $t+1$, $B_{1,t+1}$ est alors plus grand que $B_{2,t+1}$, le bras 1 est donc sélectionné.

- Le taux de détection de chaque tâche : $\hat{r}_{k,t}$. On cherche à obtenir le plus possible d’essais de la tâche ayant le meilleur taux de détection.
- La confiance dans l’estimation de ce taux : on favorise les tâches qui n’ont pas été assez évaluées (telles que $T_{k,t}$ est petit) pour être certain que ce ne sont pas les plus performantes.

Pour un budget suffisamment grand, l’algorithme *UCB-classif* garantit que les tâches sous optimales ne seront évaluées qu’un ensemble de fois négligeable ($\log N$ fois sur le budget total N). La meilleure tâche est alors évaluée $N - O(\log N)$ fois. La section suivante s’attache à la démonstration de ce résultat. L’algorithme *UCB-classif* permet ainsi de sélectionner la meilleure tâche mais aussi d’obtenir un nombre d’exécutions de cette tâche élevé. Ceci permet d’effectuer ensuite un apprentissage plus performant avant de commencer à utiliser l’ICM.

3.2.3 Quelques résultats mathématiques sur l’algorithme UCB-classif

Une formalisation plus générale et détaillée de l’apprentissage supervisé est effectuée dans la section 3 du chapitre VI page 171.

Borne supérieure avec haute probabilité Soient \mathcal{D} une distribution de probabilités sur $\mathbb{R}^d \times \{0, 1\}$ et \mathcal{H} l’ensemble des classifieurs linéaires de \mathbb{R}^d . Plus précisément, pour $(x, y) \sim \mathcal{D}$, c’est à dire que x est le vecteur des valeurs de l’ensemble des features pour un essai et y est la classe de cet essai, alors si $h \in \mathcal{H}$, $h(x)$ est la

classe obtenue par h de cet essai.

On définit la perte théorique $\{0, 1\}$ d'un classifieur h par³ :

$$L_{\mathcal{D}}(h) = \mathbf{E}_{(x,y) \sim \mathcal{D}}(\mathbb{1}_{h(x) \neq y}) .$$

Soit h^* le classifieur linéaire optimal \mathcal{D} pour la perte $\{0, 1\}$:

$$h^* = \arg \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) .$$

Si $\mathcal{X} = \{(x_1, y_1), \dots, (x_T, y_T)\}$ sont T points i.i.d de $\mathbb{R}^d \times \{0, 1\}$, tirés suivant \mathcal{D} , on définit la perte empirique $\{0, 1\}$ du classifieur h par :

$$\hat{L}_{\mathcal{X}}(h) = \frac{1}{T} \sum_{t=1}^T (\mathbb{1}_{h(x_t) \neq y_t}) .$$

Notons $\hat{h}^* \in \mathcal{H}$ le meilleur classifieur linéaire sur \mathcal{X} de \mathcal{H} pour la perte empirique $\{0, 1\}$, c.à.d :

$$\hat{h}^* = \arg \min_{h \in \mathcal{H}} \hat{L}_{\mathcal{X}}(h) .$$

Theorem 1 (Vapnik, 1982). On a avec probabilité $1 - 2\delta$

$$|L_{\mathcal{D}}(h^*) - \hat{L}_{\mathcal{X}}(\hat{h}^*)| \leq 2\sqrt{\frac{d(\log(2T/d) + 1) + \log(4/\delta)}{T}}$$

où d est la dimension de l'espace des features.

Nous avons en reprenant les notations définies dans la section précédente :

$$\begin{aligned} L_{\mathcal{D}}(h^*) &= 1 - r_k^* \\ \hat{L}_{\mathcal{X}}(\hat{h}^*) &= 1 - \hat{r}_{k,t} \end{aligned}$$

et donc avec probabilité $1 - 2\delta$, pour toute tâche k :

$$|r_k^* - \hat{r}_{k,t}| \leq 2\sqrt{\frac{d(\log(2T_{k,t}/d) + 1) + \log(4/\delta)}{T_{k,t}}} .$$

Démonstration du fonctionnement de l'algorithme UCB-classif Posons $\delta = 1/N^2$.

Comme $T_{k,t} < N$, on a avec probabilité $1 - 2/N^2$:

$$|r_k^* - \hat{r}_{k,t}| \leq 2\sqrt{\frac{d(\log(2N/d) + 1) + \log(4N^2)}{T_{k,t}}} \leq \sqrt{\frac{a \log(N)}{T_{k,t}}}, \quad (\text{IV.2})$$

avec $a = 5(d + 1)$ pour N assez grand.

³Étant donné un classifieur h (comme un SVM linéaire entraîné sur un jeu de données), $L_{\mathcal{D}}(h)$ est la probabilité qu'un nouvel essai soit mal labellisé.

Ce qui donne avec probabilité $1 - 2/N^2$:

$$r_k^* \leq \hat{r}_{k,t} + \sqrt{\frac{a \log(N)}{T_{k,t}}} \leq r_k^* + 2\sqrt{\frac{a \log(N)}{T_{k,t}}}.$$

Ce qui prouve que $B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a \log(N)}{T_{k,t}}}$ est avec forte probabilité une borne supérieure de r_k^* .

Dans le cas fortement probable où ce résultat est vrai pour tout k et tout $t < N$, on demande au sujet au temps t d'effectuer la tâche sous-optimale k si $B_{*,t} \leq B_{k,t}$, où r^* est le taux de détection de la tâche optimale k^* . Ce qui donne, d'après la dernière équation :

$$r^* \leq B_{*,t} \leq B_{k,t} \leq r_k^* + 2\sqrt{\frac{a \log(N)}{T_{k,t}}},$$

Une tâche sous-optimale k est donc effectuée seulement si

$$T_{k,t} \leq 4 \frac{a \log(N)}{(r^* - r_k^*)^2}.$$

Au total, le nombre de tâches sous-optimales effectuées par le sujet est en $O(\log(N))$. La tâche optimale est donc effectuée $N - O(\log(N))$ fois et plus précisément, au moins, $N - \sum_{k \neq * } 4 \frac{a \log(N)}{(r^* - r_k^*)^2}$ fois.

D'après l'équation IV.2, l'erreur sur le taux de détection de la meilleure tâche est avec forte probabilité :

$$|r^* - \hat{r}^*| \leq \sqrt{\frac{a \log(N)}{N - \sum_{k \neq * } 4 \frac{a \log(N)}{(r^* - r_k^*)^2}}}.$$

3.3 Évaluation de performances sur un nombre faible de réalisations

L'algorithme *UCB-classif* est basé sur le principe qu'il est possible d'évaluer de façon approchée les performances d'une tâche, ou plus précisément son taux de détection, en n'utilisant qu'un petit ensemble de réalisations. En effet, l'algorithme utilise comme récompense pour avoir tiré le bras k à l'instant t une évaluation $\hat{r}_{k,t}$ du taux de détection de la tâche associée r_k^* . Ceci est d'autant plus difficile que l'évaluation $r_{k_1,t}$ et $r_{k_2,t}$ de deux tâches (k_1 et k_2) doit être comparée même si elles ont été exécutées un nombre de fois très différent⁴ ($T_{k_1,t} \ll T_{k_2,t}$ ou $T_{k_1,t} \gg T_{k_2,t}$). Cette section est consacrée au développement d'une méthode permettant une évaluation plus précise des performances d'une tâche à l'aide d'un nombre de réalisations faible.

⁴En pratique il peut s'agir de comparer les taux de détection d'une tâche qui a été exécutée 8 fois à une qui a été exécutée 75 fois.

3.3.1 Utilisation d'un ensemble de features réduit

Pour éviter le phénomène de sur-apprentissage quand peu d'essais de chaque classe sont présents, le nombre de features doit être réduit au minimum. Basée sur les expériences préliminaires réalisées, nous avons utilisé 12 features basées sur des mesures de puissance obtenues de la façon suivante :

- 3 électrodes sont utilisées : C3 Cz et C4
- la puissance est calculée dans 2 bandes de fréquences μ et β
- 2 fenêtres sont utilisées :
 - une durant l'éventuelle imagination du mouvement
 - une postérieure pour mesurer l'ERS

Les features ne sont pas adaptées à chaque utilisateur. Il serait cependant possible, une fois qu'un nombre suffisant de réalisations d'une même tâche a été obtenu, d'augmenter le nombre de features et de les adapter au sujet voire à la tâche.

3.3.2 Essai d'utilisation de mesures de distance entre classes

Les méthodes d'apprentissage supervisé sont généralement peu adaptées quand le nombre d'exemples est extrêmement réduit. Nous avons essayé d'utiliser des mesures de distance inter-classes pour évaluer la séparation de chaque tâche à la condition de repos. De telles distances peuvent être vues comme une généralisation du t-test de Welch à des distributions multivariées. Nous avons par exemple utilisé la distance de Bhattacharyya :

$$D_B = \frac{1}{8}(M_1 - M_2)^t C^{-1}(M_1 - M_2) + \frac{1}{2} \ln \left(\frac{\det \left(\frac{C_1 + C_2}{2} \right)}{\sqrt{\det C_1 \det C_2}} \right)$$

où m_i et C_i sont la moyenne et la covariance des features de la classe i .

Bien que ces mesures aient donné des résultats satisfaisants quand le nombre d'exécutions de chaque tâche était similaire, elles n'ont pas permis une comparaison entre deux tâches ayant été exécutées un nombre de fois très différent. En effet, la distance entre une tâche ayant été exécutée une dizaine de fois et le repos dans un espace de features de dimension 12 est grandement supérieur à la distance obtenue avec un plus grand nombre d'exécutions. Cela est dû au fait que quand peu d'exemples sont présents, le bruit dans les features peut sembler discriminant.

3.3.3 Amélioration de la précision de l'évaluation sur les ensembles de données réduits

Les mesures de distance n'ayant pas donné de résultats satisfaisants, nous utilisons pour évaluer le taux de détection de chaque tâche la validation croisée. Durant les premières étapes du fonctionnement de l'algorithme, le nombre de données est extrêmement réduit et la taille des ensembles utilisés pour la validation

croisée très faible, voire peut être réduit à son minimum, c'est à dire : un exemple de chaque classe⁵. Ceci peut conduire à une évaluation très imprécise.

Exemple de fonctionnement Supposons que les ensembles de test soient constitués d'un exemple de chaque classe (soit donc un mouvement et un repos). Le taux de détection sur chaque ensemble de validation peut alors valoir :

- 100%, les deux exemples de test ont été correctement labélisés ;
- 50%, un seul des deux exemples est correct ;
- 0%, le repos a été détecté comme étant la tâche et inversement.

Dans le cas où une tâche ne peut pas être distinguée du repos, il y a quand même 25% de chance que les deux exemples soient correctement labellisés simplement par hasard. Utiliser le taux de classification minimum entre l'ensemble de test et d'apprentissage permet de diminuer la variance de l'évaluation.

Formalisation Notons $Train_i$ et $Test_i$ avec $i \in \{1..N_{CV}\}$ le taux de détection sur l'ensemble d'apprentissage et de test durant le i^{ieme} essai (ou *fold*) de la validation croisée sur une partition à N_C ensembles (voir figure VI.6 page 174). Normalement le taux de détection est évalué par :

$$r = \frac{1}{N_{CV}} \sum_{i=1}^{N_{CV}} Test_i .$$

Afin de limiter une surestimation des taux de détection lié à la chance, nous utilisons comme mesure du taux de détection :

$$r_{min} = \frac{1}{N_{CV}} \sum_{i=1}^{N_{CV}} \min(Test_i, Train_i) .$$

Ceci peut être justifié par le fait que normalement le taux de classification sur l'ensemble de test est inférieur à celui de l'ensemble d'apprentissage. Soit :

$$\forall i \in \{1..N_{CV}\}, p(Train_i \geq Test_i) \simeq 1$$

et donc $r_{min} \simeq r$. Cependant quand la taille de l'ensemble de test est très réduite, l'évaluation $Test_i$ est imprécise et il est possible que $Train_i < Test_i$. Une évaluation pessimiste mais plus précise du taux de classification est alors obtenue par r_{min} . Dans notre cas l'évaluation r_{min} permet d'obtenir une meilleure comparaison de la performance des différentes tâches.

⁵On évalue le taux de détection de chaque tâche versus le repos. Les deux classes sont donc la tâche en question et le repos.

4 ANALYSE HORS-LIGNE DE L'ALGORITHME *UCB-classif*

4.1 Objectifs et méthode pour l'analyse hors-ligne

Le but de cette section est de quantifier le gain obtenu par l'utilisation de l'algorithme *UCB-classif* pour effectuer la sélection de tâches. Pour cela, nous effectuons une comparaison hors-ligne d'*UCB-classif* à une méthode de sélection dite uniforme en simulant leur fonctionnement un grand nombre de fois sur le même jeu de données (voir le pseudo-code dans la table IV.2). C'est donc une analyse de type "bootstrap".

Simulation hors-ligne de la sélection de tâches

for $i = 1, \dots, N_{iter}$ **do**

On mélange l'ordre d'exécution des tâches qui ont été enregistrées.

for $t = 1, \dots, N$ **do**

On fournit à l'algorithme un nouvel exemple enregistré de la tâche dont il demande l'exécution.

end for

On enregistre la tâche finalement sélectionnée.

end for

On calcule la probabilité de sélectionner les différentes tâches.

Tab. IV.2 – Pseudo-code pour la simulation hors-ligne de la sélection de tâches. N est le budget accordé pour la sélection de tâches et N_{iter} le nombre de simulations.

Méthode de sélection par évaluation uniforme des tâches La méthode la plus simple pour sélectionner une tâche optimale est de demander au sujet d'effectuer chaque tâche le même nombre de fois, puis de sélectionner la plus performante. Du point de vue de la théorie des bandits, cela revient à distribuer le budget (c'est à dire le nombre de tâches que le sujet va effectuer) uniformément entre tous les bras. Cette méthode est utilisée dans [Dobrea et Dobrea, 2009, Dobrea et al., 2010].

Nécessité d'un nombre important de réalisation de chaque tâche L'avantage de l'algorithme *UCB-classif* est qu'une très grande partie du budget peut être utilisé pour évaluer la même tâche (voir la section 4.2.2). Pour pouvoir simuler son fonctionnement avec un budget N il faut donc avoir enregistré quasiment N exécutions de chacune des tâches, alors que pour la méthode uniforme $\frac{N}{K}$ (ou K est le nombre de tâches) est suffisant.

Effectuer l'enregistrement d'un grand nombre de tâches différentes avec un nombre de répétitions élevé prendrait trop de temps. Nous avons donc, pour l'analyse hors-ligne, utilisé le jeu de données de mouvements brefs enregistrés sur 10 sujets (voir section 3 du chapitre II) mais en simulant de nouvelles tâches.

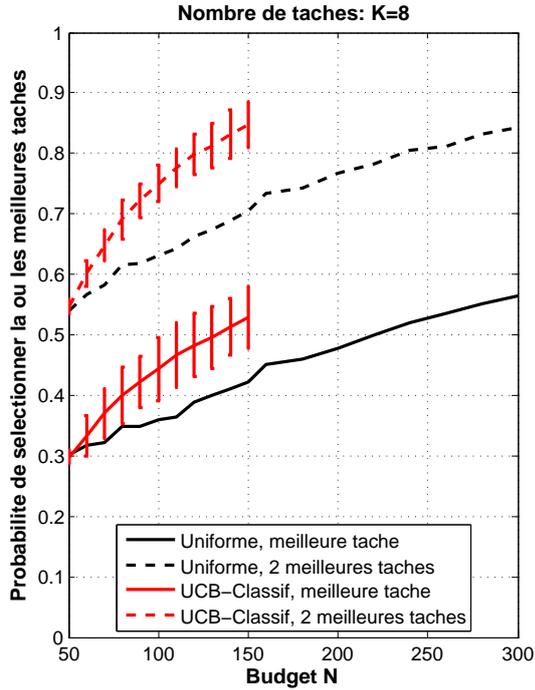


Fig. IV.3 – Évolution en fonction du budget de la probabilité de sélectionner la meilleure tâche (trait plein), ou une des 2 meilleures (pointillés), en moyenne sur l'ensemble des 10 sujets. L'algorithme *UCB-classif* (en rouge) permet une meilleure sélection que la méthode uniforme (en noir). L'intervalle de confiance à 95% sur le gain obtenu par *UCB-classif* est représenté par des barres d'erreurs.

Simulation de tâches additionnelles Nous avons créé des “tâches” additionnelles de manière artificielle en effectuant des moyennes pondérées entre les features calculées pour une tâche déjà existante et celles pour les périodes de repos. Ceci permet d'obtenir un éventail de tâches plus ou moins différenciables du repos (voir tableau IV.3).

4.2 Résultats hors-ligne

4.2.1 Amélioration du taux de sélection d'une bonne tâche

Pour certains sujets, la différence de performance entre la meilleure tâche et la seconde est très faible. Nous avons donc choisi d'afficher pour l'ensemble des sujets la probabilité de sélectionner la meilleure tâche, mais aussi l'une des deux meilleures.

La figure IV.3 donne la probabilité de sélectionner la meilleure et l'une des 2 meilleures tâches parmi 8 avec l'algorithme *UCB-classif* et la méthode uniforme. Le nombre d'exécutions de chaque tâche étant de 100 sur les données enregistrées, il n'est pas possible de simuler le fonctionnement correct de l'algorithme *UCB-classif* avec un budget supérieur à $N = 150$, l'algorithme requérant alors souvent une 101^{ème} exécution de la meilleure tâche. Inversement, l'algorithme commençant par une évaluation uniforme de toutes les tâches (afin de pouvoir effectuer la validation croisée), un budget minimum de $N = Kq$ est nécessaire⁶.

⁶Nous avons fixé $q = 6$.

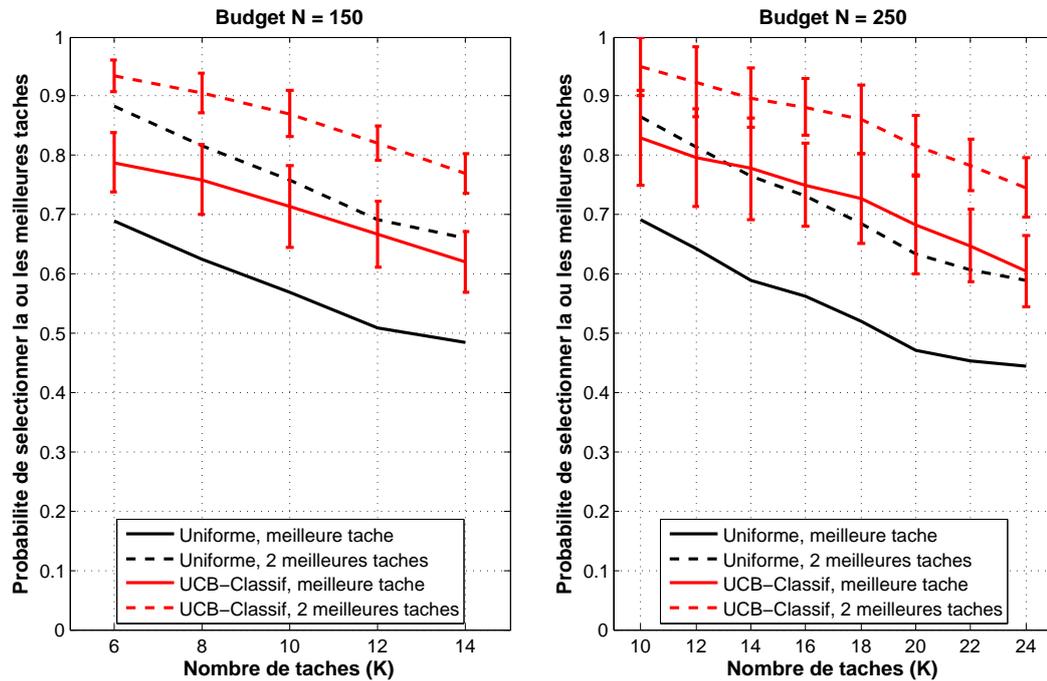


Fig. IV.4 – Évolution en fonction du nombre de tâches de la probabilité de sélectionner la meilleure tâche (trait plein), ou une des 2 meilleures (pointillés), en moyenne sur l'ensemble des 10 sujets. Le budget est fixé à $N = 150$ pour la figure de gauche et $N = 250$ pour la figure de droite ce qui correspond à des enregistrements EEG d'environ 20 et 35 minutes respectivement. L'algorithme *UCB-classif* (en rouge) permet une meilleure sélection que la méthode uniforme (en noir). L'intervalle de confiance à 95% sur le gain obtenu par *UCB-classif* est représenté par des barres d'erreurs.

Pour un budget fixé, l'algorithme *UCB-classif* permet d'augmenter grandement la probabilité de sélectionner une bonne tâche, comparé à la méthode uniforme, pour l'ensemble des sujets. De plus, il permet avec un budget de $N = 150$ d'obtenir les mêmes performances que la méthode uniforme avec un budget de $N = 300$. Pour donner un ordre de grandeur, acquérir $N = 300$ exécutions de tâche avec le protocole établi prend environ 40 minutes sans compter les temps de pause nécessaires pour que le sujet se repose. Le temps gagné grâce à l'utilisation d'*UCB-classif* est donc loin d'être négligeable.

Exploration d'un nombre de tâches plus important Les figures IV.4 donnent la probabilité de sélection d'une bonne tâche en fonction du nombre de tâches essayées pour différents budgets. L'efficacité de l'algorithme *UCB-classif* permet d'explorer un nombre de tâches plus important en gardant une durée d'expérience raisonnable.

Calcul d'un intervalle de confiance Il est important de pouvoir quantifier l'importance du gain obtenu par *UCB-classif* par rapport à la variabilité des résultats. Plutôt que d'afficher sur les figures IV.3 et IV.4 la variance inter-sujets en sus de la moyenne sur les dix sujets, nous avons choisi de calculer un intervalle de confiance à 95% du gain obtenu par *UCB-classif* comparé à la méthode uniforme. Cet intervalle est calculé, pour chaque donnée, suivant le gain obtenu pour les 10 sujets. Ceci permet de quantifier le gain obtenu par *UCB-classif* par rapport aux variations dues à la méthode employée, en s'affranchissant de la variabilité due aux performances des sujets.

Pour l'ensemble des résultats calculés, l'algorithme *UCB-classif* permet un gain positif pour chacun des 10 sujets.

4.2.2 Utilisation optimale du budget

Tâche	% de détection	Nb présentations
Main droite	88.1%	28.6 ± 12.8
Main gauche	80.5%	9.0 ± 7.5
Pieds	82.6%	11.6 ± 9.5
Langue	63.3%	4.5 ± 1.5
Pieds 80%	71.4%	5.1 ± 2.6
Pieds 60%	68.6%	4.0 ± 1.5
Pieds 40%	59.2%	3.5 ± 1.0
Pieds 20%	54.0%	3.5 ± 0.9
Budget total		70

Tab. IV.3 – Nombre moyen de présentations de chacune des $K = 8$ tâches par *UCB-classif* durant $nb_{iter} = 500$ expériences simulées. Les données ont été acquises sur un sujet suivant un protocole similaire à celui décrit dans la section 3 du chapitre III, mais avec 4 tâches. Pieds $X\%$ est un mélange des features des pieds et du repos avec un ratio $X/(100 - X)$. Le taux de détection (calculé par validation croisée sur l'ensemble des données) donne une indication sur la performance des différentes actions.

Comme le montre le tableau IV.3, l'algorithme *UCB-classif* dépense la majeure partie du budget sur les classes ayant les meilleures performances. Ceci a deux avantages :

1. Ne pas perdre de temps sur les tâches fonctionnant moins bien et ainsi mieux évaluer les bonnes tâches ce qui garantit une sélection optimale.
2. A la fin de l'expérience, la tâche finalement sélectionnée a été exécutée un grand nombre de fois⁷. Ces données constituent alors la base d'apprentissage

⁷Théoriquement, le nombre d'exécutions de la meilleure tâche est $N - \log(N)$, alors qu'il n'est que de $\frac{N}{K}$ pour la méthode uniforme.

pour les méthodes supervisées. Plus elles sont nombreuses, plus l'apprentissage pourra être efficace et meilleur sera le contrôle de l'ICM.

5 SÉLECTION D'UNE TÂCHE OPTIMALE EN-LIGNE

5.1 Scénario OpenViBE pour la sélection de tâches en-ligne

5.1.1 Objectif et principe de fonctionnement

L'objectif est de réaliser un scénario OpenViBE utilisant l'algorithme *UCB-classif* pour sélectionner la tâche d'imagination motrice la plus adaptée pour commander un bouton.

Du point de vue de l'utilisateur, le déroulement de l'expérience est similaire aux enregistrements de l'imagination de mouvements brefs. Plus précisément, des consignes de mouvements sont affichées sur l'écran de stimulation durant 2 secondes séparées par des pauses de durées variables (voir figure III.12). Cependant, l'ordre des consignes est déterminé en temps réel par l'algorithme *UCB-classif* pour sélectionner la tâche d'imagination motrice la plus appropriée.

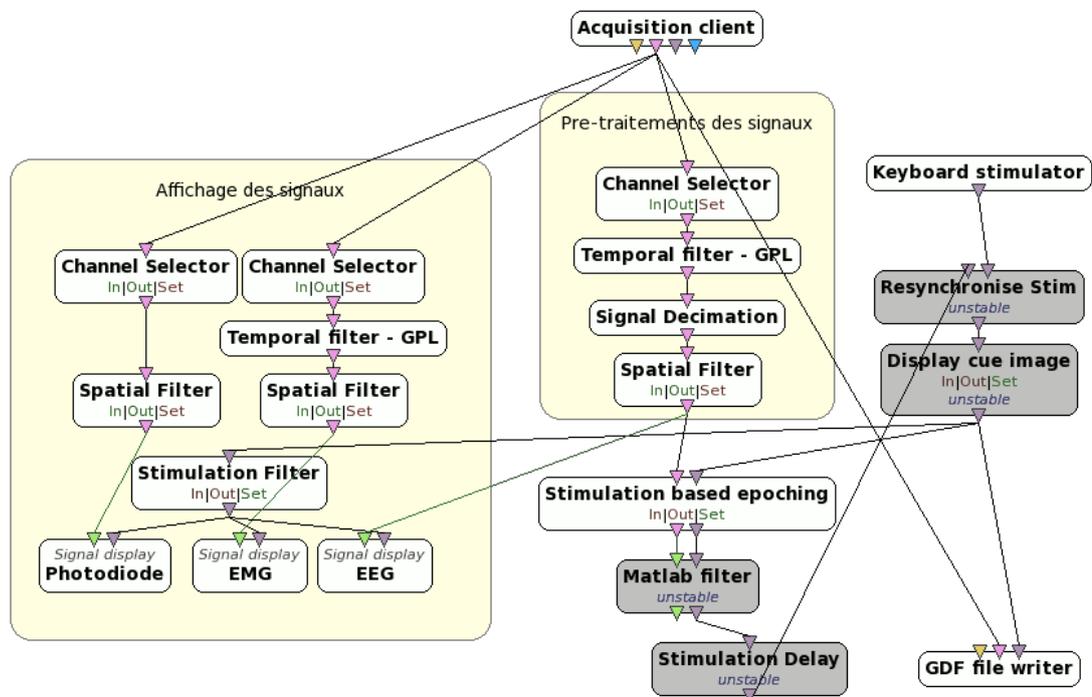


Fig. IV.5 – Scénario OpenViBE utilisé pour la sélection de tâche en-ligne.

Chaque période de signal durant laquelle l'utilisateur imagine un mouvement doit donc être analysée en temps réel pour évaluer le taux de détection de ce mouvement.

5.1.2 Détails sur le scénario OpenViBE

Le scénario OpenViBE est constitué de trois parties, qui seront détaillées successivement :

- Un affichage des signaux en temps réel pour que l'expérimentateur puisse évaluer en direct la qualité des enregistrements et la présence éventuelle d'artéfacts.
- Le traitement du signal pour évaluer en temps réel le taux de détection de chaque tâche.
- L'affichage des consignes pour indiquer à l'utilisateur quand l'imagination doit avoir lieu et avec quel mouvement.

Affichage des signaux Le signal EEG servant de base à l'extraction des features (issu des électrodes C3, Cz et C4 après l'application d'un filtre laplacien) est affiché, ainsi que les signaux EMG. Ceci permet à l'expérimentateur de vérifier que d'importants artéfacts ne soient pas présents et que le sujet n'effectue pas de contractions musculaires quand il doit seulement imaginer les mouvements.

D'autre part, le signal issu d'une photodiode permet de vérifier que l'affichage des consignes est bien synchrone avec les instants demandés. Une analyse hors-ligne automatique permet de vérifier qu'il n'y a pas de désynchronisation durant l'enregistrement.

Traitement temps réel du signal Un pré-traitement des signaux est effectué à l'aide de boîtes OpenViBE :

1. Sélection des électrodes utiles (C3, Cz et C4 ainsi que les électrodes environnantes pour le calcul du laplacien).
2. Filtrage temporel de 4 à 40Hz.
3. Ré-échantillonnage à 128 Hz par décimation.
4. Filtrage spatial de type laplacien

Pour chaque consigne d'imagination de mouvement (ou de période labellisée comme étant de repos), une fenêtre de 4 secondes de signal est découpée par la boîte "Stimulation based epoching". Cette fenêtre comprend les 2 secondes d'imagination, ainsi que la période post-imagination, durant laquelle doit se produire l'ERS. L'extraction des features et la classification sont effectuées en Matlab grâce à la boîte "Matlab filter".

Production des consignes L'affichage des consignes est effectué par la boîte "Display cue image" que nous avons réalisée. Le début de l'expérience, soit la demande

de l'affichage de la première consigne, est déclenché par l'expérimentateur⁸ grâce à la boîte "keyboard stimulator". Le choix de la consigne suivante est effectué dans la boîte "Matlab filter" par l'algorithme *UCB-classif* basé sur les taux de détection des différentes tâches⁹. Si le nombre de périodes de repos est insuffisant par rapport au nombre de répétitions de chaque tâche, la période inter-stimulations est allongée pour inclure une nouvelle période de repos.

Pour des raisons techniques liées à la gestion des timings des stimulations dans OpenViBE¹⁰ nous avons dû créer deux nouvelles boîtes "Stimulation delay" et "Resynchronise Stim" permettant respectivement d'introduire un délai sur les stimulations et de fusionner deux flux de stimulations avec une gestion paramétrable en cas de flux non mis à jour.

5.2 Résultats

Afin de vérifier le bon fonctionnement de l'algorithme *UCB-classif* en-ligne, nous avons réalisé plusieurs expériences de sélection de tâches avec quatre sujets. Les expériences devant être répétées plusieurs fois au cours d'une même session, nous sommes contraints d'utiliser un budget ($N = 60$) relativement faible et nous avons donc choisi d'explorer un nombre de tâches réduit, à savoir la main droite, les pieds et la langue.

Les résultats de la sélection pour l'ensemble des sujets sont donnés dans le tableau IV.4.

Sujet S1 La méthode uniforme a conduit à la sélection de la main droite alors que l'algorithme *UCB-classif* a sélectionné trois fois de suite les pieds. D'après l'expérience réalisée précédemment sur ce sujet, les pieds semblent être une tâche plus facile à détecter¹¹.

Sujet S6 Ce sujet a participé à plusieurs expériences qui ont permis de mettre en place le protocole décrit dans la dernière partie du chapitre III. Lors de la dernière expérience, dont les résultats sont rapportés page 107, les pieds permettent d'obtenir de meilleures performances que la main droite, cependant dans une expérience précédente ce résultat est inversé¹². Il est donc difficile de déterminer quelle est la tâche la plus performante entre la main droite et les pieds, toutefois, l'imagination de la langue est clairement moins performante.

⁸L'expérimentateur peut aussi mettre en pause l'expérience et la relancer à loisir.

⁹Nous avons modifié la boîte "Matlab filter" pour qu'elle puisse produire des stimulations.

¹⁰L'enchaînement de la boîte "Stimulation based epoching" et "Matlab filter" pose des problèmes pour la mise à jour du flux de stimulations en l'absence de données.

¹¹Quand un grand nombre de features sont utilisées nous avons obtenus des taux de classification de 95% et 86% pour les pieds et la main droite, voir page 107.

¹²Taux de détection hors-ligne obtenus pour la main droite, les pieds et la langue : 88%, 83% et 63% respectivement.

Sujet S1	Main droite		Pieds		Langue		Tâche sélectionnée
	Nb	rate	Nb	rate	Nb	rate	
Uniforme	20	79%	20	75%	20	57%	Main droite
<i>UCB-classif</i>	23	77%	27	79%	10	48%	Pieds
<i>UCB-classif</i>	16	70%	34	88%	10	74%	Pieds
<i>UCB-classif</i>	21	75%	30	82%	9	61%	Pieds

Sujet S6	Main droite		Pieds		Langue		Tâche sélectionnée
	Nb	rate	Nb	rate	Nb	rate	
<i>UCB-classif</i>	24	79%	15	56%	21	64%	Main droite
<i>UCB-classif</i>	42	94%	11	67%	7	63%	Main droite
<i>UCB-classif</i>	35	87%	14	69%	11	54%	Main droite
<i>UCB-classif</i>	26	84%	21	61%	13	50%	Main droite
<i>UCB-classif</i>	32	78%	17	74%	11	43%	Main droite

Sujet S8	Main droite		Pieds		Langue		Tâche sélectionnée
	Nb	rate	Nb	rate	Nb	rate	
<i>UCB-classif</i>	26	88%	20	86%	14	78%	Main droite
<i>UCB-classif</i>	25	78%	23	76%	12	69%	Main droite
<i>UCB-classif</i>	22	63%	26	70%	12	60%	Pieds
<i>UCB-classif</i>	27	89%	22	84%	11	71%	Main droite
<i>UCB-classif</i>	23	80%	24	83%	13	57%	Pieds

Sujet S9	Main droite		Pieds		Langue		Tâche sélectionnée
	Nb	rate	Nb	rate	Nb	rate	
<i>UCB-classif</i>	16	80%	28	97%	16	78%	Pieds
<i>UCB-classif</i>	19	72%	30	87%	11	40%	Pieds
<i>UCB-classif</i>	13	70%	30	86%	11	57%	Pieds
<i>UCB-classif</i>	15	64%	34	88%	11	55%	Pieds
<i>UCB-classif</i>	19	73%	27	86%	14	76%	Pieds

Tab. IV.4 – Tâches sélectionnées au cours des différentes expériences avec la méthode uniforme ou l’algorithme *UCB-classif*. “Nb” représente le nombre d’exécutions de chaque tâche et “Taux” l’évaluation du taux de détection à la fin de l’expérience.

Lors de l'expérience de sélection de tâche l'algorithme *UCB-classif* a permis de sélectionner, lors des cinq essais, la main droite. Une expérience additionnelle a permis de sélectionner une nouvelle fois la main droite parmi un ensemble de 5 tâches (main droite, main gauche, pieds, langue, préhension d'un objet) avec un budget fixé à $N = 120$.

Sujet S8 Pour ce sujet, l'algorithme *UCB-classif* donne lieu à la sélection de la main droite trois fois et des pieds deux fois. Lors de l'expérience précédente ces deux tâches ont permis d'obtenir d'excellentes performances avec 97% de taux de détection pour la main droite et 91% pour les pieds.

Sujet S9 L'algorithme *UCB-classif* a sélectionné cinq fois les pieds, qui pour ce sujet permettent d'obtenir des performances très légèrement meilleures que la main droite lors de l'expérience précédente.

6 CONCLUSION ET PERSPECTIVES

La sélection de tâches adaptées aux utilisateurs est un aspect important des ICM [Dobrea et Dobrea, 2009, Dobrea *et al.*, 2010]. À notre connaissance, nous sommes les premiers à avoir développé une méthode automatique et efficace pour résoudre le problème de la sélection de tâches adaptées aux utilisateurs.

Notre méthode a été appliquée avec succès au cas de la sélection d'une tâche d'imagination motrice optimale pour la commande d'un bouton. Ce travail est directement généralisable à des tâches cognitives autres que l'imagination motrice et à tout type d'ICM utilisant une seule tâche.

Extension à la sélection d'une paire de tâches optimale La généralisation à la sélection de plusieurs tâches nécessite une modification de l'algorithme. Dans la continuité du travail que nous avons réalisé, il serait particulièrement intéressant de pouvoir sélectionner une paire de tâches pour l'utilisation de deux boutons commandés par la pensée. L'objectif est alors de sélectionner deux tâches ayant chacune un taux de détection élevé ainsi qu'un taux de différenciation entre elles maximum.

Pour atteindre cet objectif, il serait possible de modifier l'algorithme *UCB-classif* de la façon suivante :

- On utilise un nombre de bras K égal au nombre de paires de tâches.
- L'évaluation de chaque bras est donnée par la combinaison du taux de détection de chacune des deux tâches et du taux de différenciation.
- Quand on sélectionne un bras, une seule des deux tâches est exécutée et l'évaluation de tous les bras ayant cette tâche dans le couple qu'il représente est mise à jour.

Reconstruction de sources pour les ICM

LE problème majeur des ICM non-invasives aujourd'hui est la faiblesse du taux de transfert d'information. Cette faiblesse provient principalement de la qualité du signal enregistré par EEG : il a un rapport signal sur bruit très faible et une résolution spatiale très limitée. Une solution envisageable pour pallier ce problème est de reconstruire l'activité cérébrale à partir du signal EEG et de se servir de cette reconstruction comme base pour la commande d'ICM. Cependant, il n'est pas encore certain qu'il soit possible de reconstruire de façon suffisamment fidèle l'activité cérébrale en temps réel pour augmenter réellement les performances des ICM.

L'équipe Athéna ayant une forte expertise de la modélisation du problème direct¹, nous avons travaillé durant le début de cette thèse sur l'utilisation de modèles réalistes pour la reconstruction de l'activité corticale en temps réel pour les ICM.

¹La résolution du problème direct est le premier pas nécessaire à la reconstruction de l'activité cérébrale

Sommaire

1	Pourquoi reconstruire l'activité des sources corticales ?	139
1.1	Augmenter la résolution spatiale de l'EEG	139
1.2	Obtenir des caractéristiques ayant un sens physiologique	140
2	Des sources aux capteurs : le problème direct	140
2.1	Modélisation de l'activité corticale	141
2.2	Modélisation anatomique	142
2.3	Résolution du problème direct	143
3	Des capteurs aux sources : le problème inverse	144
3.1	Résolution par Minimum-Norm	144
3.2	Résolution par filtrage : le "beamformer"	146
3.2.1	<i>Principe de l'approche par filtrage</i>	<i>146</i>
3.2.2	<i>Filtre idéal</i>	<i>146</i>
3.2.3	<i>Filtre de covariance minimale</i>	<i>147</i>
3.2.4	<i>Utilisation du beamformer</i>	<i>149</i>
4	Utilisation de la reconstruction de l'activité corticale pour les ICM	151
4.1	Modification de la chaîne de traitement d'une ICM	151
4.2	Sélection de features	152
4.3	Coût de l'utilisation de la reconstruction de sources	153
5	Résultats	154
5.1	Comparaison des méthodes sur des données d'imagination motrice	154
5.1.1	<i>Résultats sujet A1</i>	<i>154</i>
5.1.2	<i>Résultats sujet A2</i>	<i>155</i>
5.1.3	<i>Localisation des sources pour le sujet A2</i>	<i>155</i>
5.2	Discussion	156
5.2.1	<i>Difficultés pour l'interprétation des résultats</i>	<i>156</i>
5.2.2	<i>Méta analyse de la reconstruction de l'activité cérébrale pour les ICM</i>	<i>157</i>
5.2.3	<i>Vers une "BCI competition" permettant la reconstruction des sources</i>	<i>158</i>
6	Conclusion	159

1 POURQUOI RECONSTRUIRE L'ACTIVITÉ DES SOURCES CORTICALES POUR LES ICM ?

1.1 Augmenter la résolution spatiale de l'EEG

Les différentes méthodes d'enregistrement de l'activité cérébrale pouvant être utilisées pour réaliser une ICM ont été brièvement décrites dans la section 3 du chapitre I. Actuellement, seules l'EEG, l'ECOG et les électrodes implantées dans le cortex, sont suffisamment portables pour être utilisées à l'extérieur d'un laboratoire et permettent une mesure en temps réel de l'activité cérébrale. L'ECOG étant invasif, relativement peu d'études portant sur des ICM l'utilisent comme méthode d'enregistrement². Cependant, la plupart de ces études rapportent la possibilité de réaliser des ICM avec des périodes d'apprentissage extrêmement réduite par rapport aux ICM utilisant l'EEG [Hill *et al.*, 2006, Schalk *et al.*, 2008]. Les signaux enregistrés en EEG sont plus bruités que ceux provenant de l'ECOG et bien moins précis que ceux obtenus avec des électrodes implantées³, mais l'EEG a l'énorme avantage d'être non-invasif. Les limitations de l'EEG viennent en grande partie du fait que les signaux électriques doivent traverser le crâne avant d'atteindre les capteurs. Le crâne étant peu conducteur, il atténue grandement les signaux mesurés et agit de surcroît comme un filtre passe-bas spatial. Cela signifie que le signal mesuré par une électrode provient d'une aire spatialement étendue du cortex et pas seulement de la région se trouvant juste sous l'électrode. Cette faible résolution spatiale de l'EEG est un handicap majeur pour la réalisation d'ICM rapides et fiables. En effet, ceci limite les capacités du transducteur à différencier de façon robuste des activités cérébrales différentes.

Plusieurs techniques ont été développées pour augmenter la résolution de l'EEG. L'état de l'art des plus utilisées dans le cas des ICM a été effectué dans la section 3 du chapitre II (page 58). Les méthodes de reconstruction de sources sont peut-être les plus prometteuses puisqu'elles permettent idéalement de reconstruire l'activité de chaque source cérébrale grâce à l'utilisation de modèles anatomiques. Idéalement, ceci permettrait d'approcher avec des ICM non-invasives le niveau de contrôle d'ICM utilisant l'ECOG voire des électrodes implantées. Cependant, à cause du nombre limité de capteurs et du rapport signal sur bruit extrêmement faible, il n'est pas certain qu'il soit possible de reconstruire, en temps réel, l'activité cérébrale avec suffisamment de précision pour réellement améliorer les performances des ICM.

²L'utilisation de l'ECOG chez l'être humain est généralement réservé au diagnostic épileptique pré-opératoire. Des ICM utilisant l'ECOG ont cependant été étudiées entre la pose des électrodes et l'intervention chirurgicale.

³Les électrodes implantées, étant extrêmement invasives, sont utilisées quasi exclusivement chez l'animal.

1.2 Obtenir des caractéristiques ayant un sens physiologique

La reconstruction de l'activité cérébrale permet d'obtenir un signal ayant un sens physiologique. De la même façon que l'on espère augmenter la précision de l'EEG en reconstruisant l'activité des sources cérébrales, on peut espérer remonter à l'origine physiologique du signal et ainsi préciser le sens que l'on peut donner aux caractéristiques.

On sait par exemple que la variation de puissance dans le μ et le β sur les électrodes centrales est liée aux activités sensori-motrices. De façon plus précise, on peut lier l'activité d'une source du cortex sensori-moteur avec les différentes parties du corps suivant dans quelle aire elle se situe (voir figure I.24 page 43 et le chapitre III).

La nature physiologique des sources cérébrales permet de se servir de connaissances cognitives pour améliorer les ICM. Plus précisément, il s'agit d'utiliser des connaissances *a priori* pour améliorer le transducteur. Par exemple, pour une ICM utilisant l'imagination motrice il serait pertinent de se limiter à l'utilisation de sources situées dans le cortex sensori-moteur, voire dans les régions associées aux tâches effectuées.

Inversement, les ICM utilisant la reconstruction de l'activité cérébrale permettent d'obtenir de nouvelles connaissances sur le fonctionnement du système nerveux. Par exemple, l'analyse des cartes temps-fréquences des mouvements brefs réalisée dans le chapitre III laisse supposer des mécanismes complexes régissant les rythmes sensori-moteurs μ et β . En particulier, on peut se demander si les réseaux neuronaux à l'origine du rebond β sont identiques, et situés dans les mêmes régions, que ceux qui se désynchronisent durant le mouvement. Malheureusement, la précision spatiale de l'EEG brut est trop faible pour répondre à de tels questionnements. La reconstruction de l'activité cérébrale pourrait ainsi permettre une avancée de la recherche sur les mécanismes neuronaux sous-jacents au fonctionnement des ICM.

2 DES SOURCES AUX CAPTEURS : LE PROBLÈME DIRECT

Pour effectuer la reconstruction de l'activité cérébrale en fonction des mesures effectuées sur les capteurs situés à l'extérieur de la tête, il est nécessaire d'effectuer une modélisation des sources cérébrales ainsi que des différents milieux que le signal doit traverser. Ces modèles physiques permettent de calculer la contribution théorique de chaque source cérébrale aux mesures sous forme d'une matrice de gain (\mathbf{G}), ce que l'on nomme le problème direct [Baillet *et al.*, 2001]. La reconstruction de l'activité cérébrale consiste à résoudre un problème inverse où l'on cherche

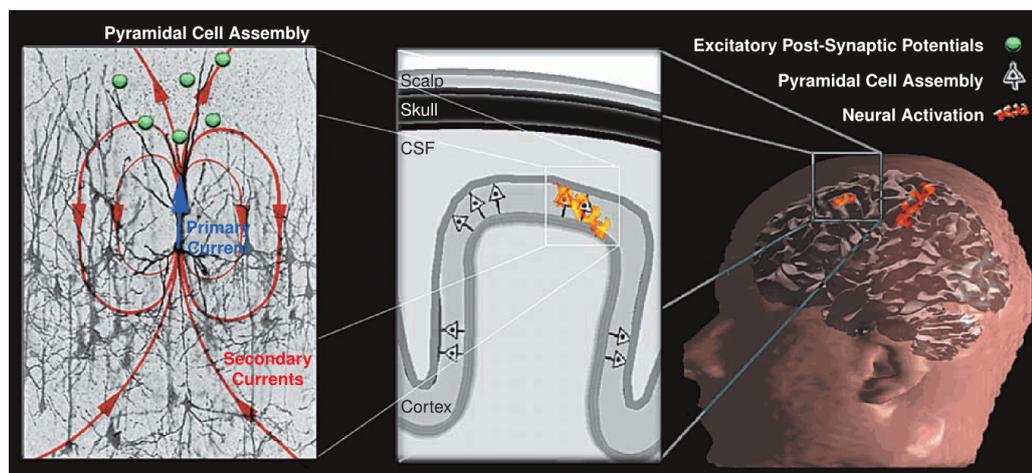


Fig. V.1 – Modélisation des colonnes fonctionnelles corticales par des dipôles (Image reprise de [Baillet et al., 2001]).

les sources compatibles avec les mesures au travers de la relation représentée par la matrice de gain \mathbf{G} .

Cette section commence par la description des modèles utilisés pour représenter les sources corticales, puis décrit les différents modèles de tête qui peuvent être utilisés et enfin expose brièvement le problème direct.

2.1 Modélisation de l'activité corticale

Les mesures effectuées en EEG proviennent principalement de l'activation de colonnes fonctionnelles pouvant regrouper plusieurs centaines de milliers de neurones corticaux (voir section 2 du chapitre I page 19). Ces colonnes, mesurant quelques millimètres de diamètre, sont orientées perpendiculairement à la surface du cortex et peuvent être modélisées par des dipôles (voir figure V.1).

Comme expliqué dans la section 3.3.2 du chapitre II (page 62), il existe deux approches de modélisation et reconstruction de l'activité cérébrale :

- l'approche de type localisation de dipôles (ou *dipole fit*) qui consiste à modéliser la majeure partie de l'activité par un faible nombre de dipôles (entre 1 et 5 généralement).
- l'approche de type sources distribuées⁴, où l'activité corticale est modélisée par un très grand nombre de dipôles (jusqu'à plusieurs dizaines de milliers de sources).

⁴La reconstruction de sources par filtrage, comme le beamformer, fait partie des méthodes de sources distribuées. En effet, même si elle permet d'estimer l'activité d'une seule source, on suppose que l'activité corticale est due à un très grand nombre de sources.

Dans le cas des sources distribuées, plusieurs modèles peuvent être utilisés pour modéliser l'activité cérébrale :

- Modèle volumique : l'activité cérébrale est modélisée par une grille volumique de dipôles occupant l'ensemble du cerveau. Il faut alors estimer l'activité et l'orientation de l'ensemble des dipôles.
- Modèle surfacique. Les mesures effectuées en EEG étant en très grande partie dues à l'activité corticale, on modélise le cortex par une surface sur laquelle on place des dipôles. Il existe deux types de modèles de dipôles surfaciques :
 - Modèle à orientation non contrainte : il s'agit comme pour le modèle surfacique d'estimer l'activité et l'orientation des dipôles. Une surface modélisant le cortex de façon grossière peut être utilisée (voir figure V.3, image du milieu).
 - Modèle à orientation contrainte : les colonnes de neurones pyramidaux étant orthogonales au cortex (voir figure V.1), on contraint l'orientation des dipôles à être normale à la surface modélisant le cortex. Cette surface doit alors être suffisamment précise pour représenter les sillons du cortex (voir figure V.2).

2.2 Modélisation anatomique

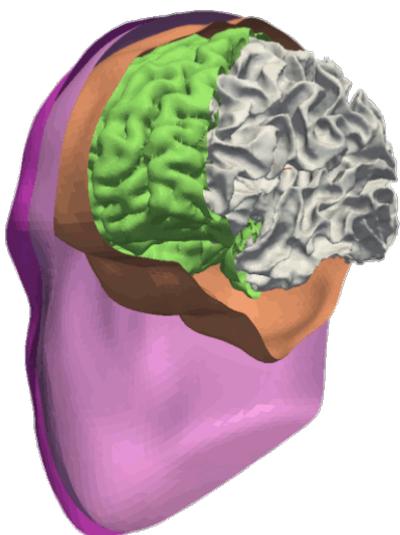


Fig. V.2 – *Modèle de tête réaliste détaillé basé sur la segmentation de l'IRM du sujet A2 : [fuchsia] scalp [mauve] crâne [orange] liquide céphalorachidien [vert] matière grise [gris] matière blanche. Seules les 3 premières surfaces sont utilisées pour le modèle anatomique. La surface verte ou grise est utilisée pour modéliser les sources distribuées (une surface plus lisse est utilisée dans le cas de sources ayant une orientation non contrainte). (Image reprise de [Olivi, 2011].)*

Les milieux séparant les sources des capteurs ont des conductivités électriques très différentes. Pour obtenir un modèle direct fidèle, il est donc nécessaire d'effectuer un modèle anatomique des différents milieux et en particulier du crâne, qui à une conductivité bien inférieure au scalp et au cerveau. La plupart des modèles anatomiques sont donc constitués de trois couches : le scalp, le crâne et l'intérieur du crâne (englobant le liquide céphalo-rachidien et le cerveau). Plusieurs niveaux

de précision anatomique peuvent être utilisés pour modéliser la tête du sujet :

1. Le modèle le plus grossier est constitué de trois sphères concentriques. Il a comme avantage sa simplicité et permet le calcul d'une solution analytique. Ce type de modèle est habituellement utilisé en MEG mais moins en EEG car il ne permet pas une localisation très précise⁵.
2. Un modèle commun d'un crâne et d'un cerveau moyen peut être utilisé sans qu'il soit adapté au sujet [**Grave de Peralta Menendez et al., 2005**].
3. Pour obtenir des modèles plus adaptés aux sujets il est nécessaire de construire le modèle anatomique à partir d'images IRM du patient.

La section 3.3.2 du chapitre II (page 62) recense quelques utilisations de ces différents modèles dans le cadre des ICM.

Nous avons utilisé un modèle réaliste extrait à partir d'une IRM de la tête du sujet A2 (voir figures V.2 et V.3).

2.3 Résolution du problème direct

Le problème direct consiste à calculer la contribution théorique de chaque source sur chaque capteur en fonction de sa position et son orientation. On note \mathbf{G} la matrice de gain où $G_{i,j}$ est l'activité mesurée par le $i^{\text{ème}}$ capteur pour une activité unitaire de la $j^{\text{ème}}$ source. La contribution théorique des sources aux capteurs est donc calculée par :

$$\tilde{\mathbf{m}}(t) = \mathbf{G}\mathbf{s}(t)$$

Le champ électrique étant particulièrement sensible aux variations de conductivité, il est important d'utiliser un modèle anatomique précis et une bonne résolution du problème direct pour obtenir une reconstruction de source fiable en EEG.

Nous utilisons *OpenMEEG* [**Kybic et al., 2005, Gramfort et al., 2011**] pour calculer la matrice de gain à partir du modèle anatomique, d'une distribution de source et de la position des électrodes. Plus précisément le calcul de \mathbf{G} est effectué par une méthode utilisant des éléments finis surfaciques pour résoudre les équations de Maxwell⁶.

⁵Un modèle sphérique a tout de même été utilisé dans plusieurs études portant sur les ICM [**Kamoussi et al., 2005, Noirhomme et al., 2008, Lotte et al., 2008a, Grosse-Wentrup et al., 2009, Ahn et al., 2012**].

⁶Cette méthode permet de résoudre le problème direct en utilisant uniquement des maillages des interfaces des différents milieux (il n'est donc pas nécessaire de mailler les volumes entre celles-ci).

3 DES CAPTEURS AUX SOURCES : LE PROBLÈME INVERSE

La matrice de gain \mathbf{G} , obtenue par la résolution du problème direct, permet de lier l'activité des sources, en chaque instant, aux mesures théoriques établies sur les capteurs :

$$\tilde{\mathbf{m}}(t) = \mathbf{G}\mathbf{s}(t)$$

Idéalement, il suffirait d'inverser la matrice \mathbf{G} pour reconstruire l'activité des sources à partir des mesures. Deux points rendent cependant la résolution du problème inverse beaucoup plus difficile :

- Le nombre de sources N_S , dont on souhaite reconstruire l'activité, est bien supérieur au nombre de capteurs (ou électrodes) N_E . Il s'agit donc d'un problème inverse **sous-déterminé**⁷.
- La présence de bruit. Les mesures effectuées sur les capteurs sont un mélange de l'activité corticale et de bruit.

Sources de bruits en EEG

- *Bruits électromagnétiques ambiants dus à des appareils électriques. Le courant électrique 220V servant pour l'alimentation crée des perturbations à 50Hz bien supérieures au signal d'intérêt.*
- *Bruits provenant de l'activité physiologique du sujet. Il s'agit principalement de clignements des yeux et des serremments des mâchoires⁸. La contraction de muscles proches des électrodes crée des artefacts bien plus puissants que l'activité corticale.*
- *Bruits intrinsèques aux capteurs.*

Plusieurs méthodes permettent de résoudre le problème inverse de façon approchée. Nous avons utilisé deux méthodes, issues d'approches totalement différentes, qui peuvent être appliquées en temps réel. Ces deux méthodes font l'objet des deux prochaines sections.

3.1 Résolution par Minimum-Norm

On suppose que les mesures effectuées sur les capteurs sont constituées de la contribution de l'activité des sources corticales auxquelles s'ajoute un bruit, soit :

$$\mathbf{m}(t) = \mathbf{G}\mathbf{s}(t) + \mathbf{n}(t)$$

Le modèle de bruit additif est adapté pour modéliser les bruits électromagnétiques extérieurs ainsi que ceux dus à l'activité musculaire, ce qui n'est pas forcément le cas des bruits intrinsèques aux capteurs.

⁷La matrice \mathbf{G} est rectangulaire et n'est donc pas inversible.

⁸Les mouvements oculaires, les contractions au niveau du cou ainsi que les battements cardiaques créent aussi des perturbations.

La résolution par Minimum-Norm consiste à chercher une distribution de l'activité des sources qui explique au mieux au sens des moindres carrés les mesures effectuées, soit :

$$\min_{\mathbf{S}} \|\mathbf{M} - \mathbf{GS}\|_2^2$$

où $\mathbf{M} = [\mathbf{m}(t_1) \dots \mathbf{m}(t_2)]$ et $\mathbf{S} = [\mathbf{s}(t_1) \dots \mathbf{s}(t_2)]$.

Le nombre de sources étant bien supérieur au nombre de capteurs, la solution n'est pas forcément unique et est instable par rapport à $\mathbf{m}(t)$. Une méthode pour obtenir une solution unique et stable est d'utiliser la régularisation de Tikhonov :

$$\min_{\mathbf{S}} \|\mathbf{M} - \mathbf{GS}\|_2^2 + \lambda \|\mathbf{S}\|_2^2 \quad (\text{V.1})$$

où λ est un paramètre de régularisation qui doit être ajusté.

L'avantage de cette régularisation est qu'une solution analytique de (V.1) peut être calculée :

$$\mathbf{S}_\lambda = \mathbf{A}_\lambda \mathbf{M}$$

avec $\mathbf{A}_\lambda = \mathbf{G}^T (\mathbf{G}\mathbf{G}^T + \lambda \mathbf{I})^{-1}$.

La matrice \mathbf{A}_λ ne dépendant que de \mathbf{G} et λ on a :

$$\mathbf{s}_\lambda(t) = \mathbf{A}_\lambda \mathbf{m}(t)$$

La reconstruction de source par Minimum-Norm revient donc à chercher la distribution de sources qui explique au mieux l'ensemble des mesures tout en étant de norme minimale. Cette approche a été utilisée plusieurs fois dans le cadre des ICM [Besserve, 2007, Babiloni *et al.*, 2007, Noirhomme *et al.*, 2008, Fruitet et Clerc, 2009, Fruitet et Clerc, 2010, Besserve *et al.*, 2011].

Importance du paramètre de régularisation Si le paramètre de régularisation est trop faible, les mesures reconstruites à partir de la solution $\mathbf{S}_\lambda : \widetilde{\mathbf{M}}_\lambda = \mathbf{GS}_\lambda$, vont être extrêmement proches des mesures effectuées \mathbf{M} . Cela veut dire que l'on aura projeté une partie importante du bruit \mathbf{N} sur les sources. Théoriquement, la distribution du bruit sur les capteurs est difficilement explicable par l'activité des sources au travers de la matrice de gain \mathbf{G} . Expliquer le bruit demanderait donc une activité des sources importante ce qui devrait être limité par le terme de régularisation si le paramètre λ est bien ajusté.

Choix de la norme pour la régularisation Le choix de la norme 2 pour la régularisation permet d'obtenir une méthode de reconstruction de source linéaire. Une fois la matrice \mathbf{A}_λ calculée, la reconstruction de l'activité corticale à partir des mesures se fait par une simple multiplication matricielle et peut donc aisément être calculée en temps réel.

L'utilisation de la norme 2 produit une solution anormalement lisse⁹, alors que la norme 1 permet d'obtenir une solution plus contrastée. Il est même possible

⁹Ceci est dû au fait que la norme 2 pénalise fortement les sources ayant de fortes activités.

d'utiliser une pseudo norme $\|\cdot\|_p$, avec $0 \leq p < 1$, pour obtenir une solution parcimonieuse. Cependant les algorithmes d'optimisation de la pseudo-norme $\|\cdot\|_{p \leq 1}$ sont coûteux en temps et seule la norme 2 permet une résolution facilement calculable en temps réel.

Minimum-Norm pondéré Les sources plus éloignées des capteurs ont une contribution plus faible aux mesures. La recherche d'une solution de norme minimale a donc tendance à favoriser les sources plus proches des capteurs. Pour éviter la sous-évaluation des sources profondes on applique une matrice de poids \mathbf{P} diagonale aux sources avant leur pénalisation :

$$\min_{\mathbf{S}} \|\mathbf{M} - \mathbf{G}\mathbf{S}\|_2^2 + \lambda \|\mathbf{P}\mathbf{S}\|_2^2$$

où $P_{j,j} = \sqrt{\sum_{i=1}^{N_E} (G_{i,j})^2}$.

La solution est alors :

$$\mathbf{S}_\lambda = \mathbf{A}_\lambda \mathbf{M}$$

avec $\mathbf{A}_\lambda = \mathbf{P}^{-2} \mathbf{G}^T (\mathbf{G} \mathbf{P}^{-2} \mathbf{G}^T + \lambda \mathbf{I})^{-1}$.

3.2 Résolution par filtrage : le "beamformer"

3.2.1 Principe de l'approche par filtrage

Les approches de type Minimum-Norm cherchent à trouver l'activité de sources expliquant au mieux les mesures effectuées. Un de leurs inconvénients est qu'il y a un risque de projeter les bruits extérieurs sur les sources. Le beamformer, parfois appelé "formation de voies", est le résultat d'une approche complètement différente consistant à séparer l'information pertinente du reste par filtrage. Plus précisément il s'agit de construire un filtre spatial ne laissant passer que l'activité provenant d'une région d'intérêt. Cette méthode a été développée originellement dans le domaine du traitement de réseaux de capteurs (*array signal processing*) qui incluent les radars sonars et l'exploration sismique [Capon, 1969, Van Veen et Buckley, 1988].

3.2.2 Filtre idéal

Nous avons défini, dans le chapitre II, un filtre spatial linéaire $\mathbf{w} = [w_1 \dots w_{N_E}]^T$ par :

$$\mathbf{w}^T \mathbf{m}(t) = \sum_{e=1}^{N_E} w_e m_e(t)$$

où $\mathbf{m}(t) = [m_1(t) \dots m_{N_E}(t)]^T$ est le vecteur des mesures effectuées à l'instant t sur l'ensemble des N_E électrodes.

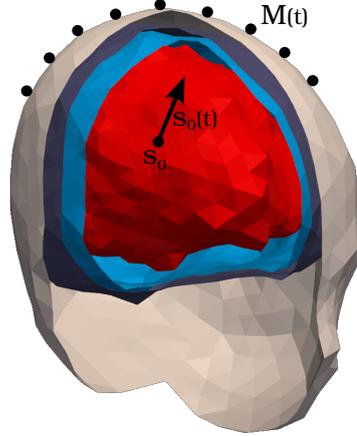


Fig. V.3 – Modèle de tête réaliste lisse basée sur la segmentation de l'IRM du sujet A2 : [beige] scalp [gris] crâne [bleu] liquide céphalorachidien [rouge] cortex. Seules les 3 premières surfaces sont utilisées pour le modèle anatomique. La surface rouge est utilisée pour modéliser les sources distribuées avec une orientation non contrainte. Le beamformer permet, à partir des mesures $\mathbf{m}(t)$, de déduire l'activité $s_0(t)$ de la source s_0 .

Notons s_0 la source corticale d'intérêt et $s_0(t)$ son activité au cours du temps. La résolution du problème direct permet de calculer la contribution unitaire de cette source sur l'ensemble des capteurs : \mathbf{g}_0 . D'une façon plus générale, notons \mathbf{g}_i la contribution unitaire d'une source s_i sur les capteurs.

Les mesures effectuées sur les capteurs ont trois origines :

- la source d'intérêt s_0 , dont la contribution aux mesures est $\mathbf{g}_0 s_0(t)$,
- d'autres sources corticales dont on modélise la contribution par $\sum_{i \neq 0} \mathbf{g}_i s_i(t)$,
- du bruit $\mathbf{n}(t)$ que l'on suppose additif.

On a donc :

$$\mathbf{m}(t) = \mathbf{g}_0 s_0(t) + \sum_{i \neq 0} \mathbf{g}_i s_i(t) + \mathbf{n}(t) \quad (\text{V.2})$$

Idéalement on cherche à construire un filtre \mathbf{w}_{s_0} passe bande infiniment fin, ne laissant passer que l'activité $s_0(t)$ de la source d'intérêt au cours du temps. Ceci se traduit par :

$$\hat{s}_0 = \mathbf{w}_{s_0}^T \mathbf{m}(t) = s_0(t) \quad (\text{V.3})$$

Il faudrait pour cela que :

$$\begin{cases} \mathbf{w}_{s_0}^T \mathbf{g}_0 = 1 \\ \mathbf{w}_{s_0}^T \mathbf{g}_i = 0 \text{ pour } i \neq 0 \\ \mathbf{w}_{s_0}^T \mathbf{n}(t) = 0 \forall t \end{cases}$$

3.2.3 Filtre de covariance minimale

Comme \mathbf{w}_{s_0} n'a au mieux que N_E degrés de liberté, il est généralement impossible d'obtenir une atténuation totale de l'ensemble des sources différentes de la source d'intérêt. C'est à dire que la condition $\mathbf{w}_{s_0}^T \mathbf{g}_i = 0$ ne peut pas être satisfaite pour toutes les sources sans avoir $\mathbf{w}_{s_0} = 0$. On peut alors chercher le filtre spatial le plus adapté en résolvant un problème de minimisation cherchant à atténuer les $\mathbf{w}_{s_0}^T \mathbf{g}_i, i \neq 0$ sans les annihiler. Cette minimisation s'effectue sous la

contrainte $\mathbf{w}_{s_0}^T \mathbf{g}_0 = 1$, qui permet de garder un gain du filtre unitaire sur la région d'intérêt. L'approche la plus répandue, nommée beamformer linéaire à variance minimale, consiste à minimiser la variance de l'estimation \hat{s}_0 de la source d'intérêt [Spencer *et al.*, 1992, Veen *et al.*, 1997, Gross et Ioannides, 1999].

Nous allons démontrer que, sous deux hypothèses, cela revient à minimiser la contribution du bruit et des autres sources corticales. Les deux hypothèses sont :

1. Le bruit est supposé indépendant de l'activité corticale.
2. Les activités des différentes sources corticales sont supposées décorrélées.

L'hypothèse de décorrélation du bruit à l'activité corticale est justifié pour les bruits extérieurs (pollution électromagnétique dû à l'environnement) et dans une moindre mesure pour les artéfacts provenant de l'activité physiologique du sujet (clignements des yeux, rythme cardiaque, contractions musculaires). Cependant, les différentes sources corticales étant fortement connectées, supposer leur indépendance est beaucoup plus critiquable.

Démonstration de l'intérêt de la minimisation de la variance de \hat{s}_0 Notons :

$\mathbf{C}_M = \langle \mathbf{m}(t)\mathbf{m}(t)^T \rangle_t$ la covariance des mesures,

$C_{s_0} = \langle s_0(t)s_0(t)^T \rangle_t$ celles de la source d'intérêt,

$\mathbf{C}_S = \langle \mathbf{S}(t)\mathbf{S}(t)^T \rangle_t$ celle des autres sources et

$\Sigma = \langle \mathbf{n}(t)\mathbf{n}(t)^T \rangle_t$ celle du bruit.

Par définition de \hat{s}_0 (V.3) et comme les coefficients du filtre n'évoluent pas avec le temps :

$$\begin{aligned} C_{\hat{s}_0} &= \langle \mathbf{w}_{s_0}^T \mathbf{m}(t)\mathbf{m}(t)^T \mathbf{w}_{s_0} \rangle_t \\ &= \mathbf{w}_{s_0}^T \mathbf{C}_M \mathbf{w}_{s_0} \end{aligned}$$

D'après l'équation (V.2) :

$$\mathbf{C}_M = \langle (\mathbf{g}_0 s_0(t) + \sum_{i \neq 0} \mathbf{g}_i s_i(t) + \mathbf{n}(t)) (\mathbf{g}_0 s_0(t) + \sum_{i \neq 0} \mathbf{g}_i s_i(t) + \mathbf{n}(t))^T \rangle_t$$

D'après les hypothèses, les sources étant décorrélées entre elles et du bruit, on a :

$$\begin{aligned} \mathbf{C}_M &= \langle \mathbf{g}_0 s_0(t)s_0(t)^T \mathbf{g}_0^T \rangle_t + \sum_{i \neq 0} \langle \mathbf{g}_i s_i(t)s_i(t)^T \mathbf{g}_i^T \rangle_t + \langle \mathbf{n}(t)\mathbf{n}(t)^T \rangle_t \\ &= \mathbf{g}_0 \langle s_0(t)s_0(t)^T \rangle_t \mathbf{g}_0^T + \sum_{i \neq 0} \mathbf{g}_i \langle s_i(t)s_i(t)^T \rangle_t \mathbf{g}_i^T + \Sigma \\ &= \mathbf{g}_0 C_{s_0} \mathbf{g}_0^T + \mathbf{G} \mathbf{C}_S \mathbf{G}^T + \Sigma \end{aligned}$$

et donc

$$\begin{aligned} \mathbf{w}_{s_0}^T \mathbf{C}_M \mathbf{w}_{s_0} &= \mathbf{w}_{s_0}^T \mathbf{g}_0 C_{s_0} \mathbf{g}_0^T \mathbf{w}_{s_0} + \mathbf{w}_{s_0}^T \mathbf{G} \mathbf{C}_S \mathbf{G}^T \mathbf{w}_{s_0} + \mathbf{w}_{s_0}^T \Sigma \mathbf{w}_{s_0} \\ &= C_{s_0} + \mathbf{w}_{s_0}^T \mathbf{G} \mathbf{C}_S \mathbf{G}^T \mathbf{w}_{s_0} + \mathbf{w}_{s_0}^T \Sigma \mathbf{w}_{s_0} \end{aligned} \quad (\text{V.4})$$

sous condition que $\mathbf{w}_{s_0}^T \mathbf{g}_0 = 1$.

Comme \mathbf{C}_S et $\mathbf{\Sigma}$ sont des matrices semi-définies positives, $\mathbf{w}_{s_0}^T \mathbf{G} \mathbf{C}_S \mathbf{G}^T \mathbf{w}_{s_0}$ et $\mathbf{w}_{s_0}^T \mathbf{\Sigma} \mathbf{w}_{s_0}$ sont positifs. Minimiser la variance $C_{s_0} = \mathbf{w}_{s_0}^T \mathbf{C}_M \mathbf{w}_{s_0}$ sous la contrainte $\mathbf{w}_{s_0}^T \mathbf{g}_0 = 1$ permet donc de minimiser la contribution du bruit et des autres sources à l'estimation $\hat{s}_0(t)$ de l'activité de la source d'intérêt s_0 . De façon plus précise, les poids du filtre sont ajustés pour filtrer d'autant plus les sources (différentes de la source d'intérêt) qui ont une contribution importantes aux mesures. C'est à dire, celles pour lesquelles $\|\mathbf{g}_i\|^2 C_{s_i}$ est important.

Solution du beamformer linéaire à variance minimale La minimisation de $\mathbf{w}_{s_0}^T \mathbf{C}_M \mathbf{w}_{s_0}$ sous la contrainte $\mathbf{w}_{s_0}^T \mathbf{g}_0 = 1$ se calcule grâce à la méthode des coefficients de Lagrange et a pour solution :

$$\mathbf{w}_{s_0} = \frac{\mathbf{C}_M^{-1} \mathbf{g}_0}{\mathbf{g}_0^T \mathbf{C}_M^{-1} \mathbf{g}_0} \quad (\text{V.5})$$

3.2.4 Utilisation du beamformer

Reconstruction de l'activité corticale par beamformer Contrairement à la résolution du problème inverse par Minimum-Norm, qui oblige à reconstruire l'ensemble de l'activité corticale, le beamformer permet de reconstruire l'activité d'une unique source corticale. Il suffit de calculer le vecteur de gain \mathbf{g}_0 par résolution du problème direct, puis un seul filtre \mathbf{w}_{s_0} . Cela revient à placer une électrode virtuelle en n'importe quel point du cortex.

Il est également possible d'estimer l'ensemble de l'activité corticale dans une région d'intérêt, ou dans tout le cortex, par beamformer en construisant un filtre par source corticale. On calcule, par le problème direct, l'ensemble de la matrice de gain \mathbf{G} et on construit N_S filtres utilisant chacun une colonne de la matrice de gain. Le beamformer a été utilisé plusieurs fois pour essayer d'améliorer les capacités des ICM [[Grosse-Wentrup et al., 2009](#), [Fruitet et Clerc, 2009](#), [Fruitet et Clerc, 2010](#), [Ahn et al., 2010](#), [Ahn et al., 2012](#)].

D'après [[Grosse-Wentrup et al., 2009](#)], le beamformer est particulièrement efficace quand les enregistrements sont pollués par de nombreux artéfacts. Ceci peut être expliqué par le fait que le beamformer, en filtrant le signal provenant des sources d'intérêt, permet théoriquement de supprimer le bruit et les artéfacts.

Résolution spatiale du beamformer Comme expliqué précédemment il n'est pas possible de construire un filtre ne laissant passer que l'activité provenant de la source d'intérêt. Le pouvoir discriminant du beamformer dépend, comme pour la reconstruction par Minimum-Norm, du nombre de capteurs et de la diffusion spatiale due au problème direct. En d'autres termes, deux sources ayant une contribution similaire sur l'ensemble des capteurs ne pourront pas être correctement séparées.

Le cas des sources corrélées Un inconvénient supplémentaire provient de l’hypothèse de non corrélation des sources. Théoriquement, si une source est corrélée avec la source d’intérêt, l’activité de la source reconstruite est atténuée et perturbée (et ce d’autant plus que la corrélation est importante). Des valeurs théoriques de l’atténuation et de la perturbation sont données dans [Van Veen *et al.*, 1997] et [Sekihara *et al.*, 2002b]. Dans [Sekihara *et al.*, 2002a] une analyse numérique montre que pour des sources ayant une corrélation inférieure à 0.6 l’atténuation est très faible, cependant cette atténuation peut avoir un effet notable sur le déroulement temporel de l’activité reconstruite au niveau d’une source.

——— Origine de l’atténuation due aux sources corrélées

Le filtre est optimisé pour minimiser la variance de l’activité reconstruite de la source d’intérêt. Si les sources sont décorrélées, cela revient à minimiser la contribution des autres sources et du bruit. Cependant, si une source s_1 est corrélée avec la source d’intérêt s_0 , le terme de couplage entre les deux sources fait apparaître un terme croisé qui peut être négatif dans l’équation (V.4) :

$$\mathbf{w}_{s_0}^T \mathbf{C}_M \mathbf{w}_{s_0} = C_{s_0} + 2\mathbf{C}_{s_0, s_1} \mathbf{g}_1^T \mathbf{w}_{s_0} + \mathbf{w}_{s_0}^T \mathbf{G} \mathbf{C}_S \mathbf{G}^T \mathbf{w}_{s_0} + \mathbf{w}_{s_0}^T \mathbf{\Sigma} \mathbf{w}_{s_0}.$$

Ce couplage va alors tendre à sous-estimer l’activité de la source d’intérêt.

Afin de se prémunir des distorsions causées par une source parasite s_1 corrélée à la source d’intérêt s_0 on peut ajouter comme contrainte que le filtre doit annuler le signal provenant de cette source, soit $\mathbf{w}_{s_0}^T \mathbf{g}_1 = 0$. Il faut alors calculer, par résolution du problème direct, les contributions \mathbf{g}_0 et \mathbf{g}_1 des deux sources sur les capteurs. Dans [Dalal *et al.*, 2006] cette approche est généralisée au cas d’un ensemble de sources dont on veut supprimer la contribution.

Amélioration du beamformer dans le cas de rapport signal sur bruit faible Un des problèmes du beamformer de variance minimale est sa grande sensibilité à la qualité du modèle direct \mathbf{g}_0 et aux erreurs d’estimations de la matrice de covariance des mesures \mathbf{C}_M . Ceci est dû à l’inversion de la matrice \mathbf{C}_M et au fait que la méthode est non régularisée.

Des modifications ont été apportées au beamformer pour le rendre plus robuste. La plus simple consiste à améliorer le conditionnement de la matrice \mathbf{C}_M en utilisant une régularisation : $\hat{\mathbf{C}}_M = \mathbf{C}_M + \lambda \mathbf{I}$ [Gross et Ioannides, 1999]. Le filtre devient alors :

$$\mathbf{w}_{s_0} = \frac{(\mathbf{C}_M + \lambda \mathbf{I})^{-1} \mathbf{g}_0}{\mathbf{g}_0^T (\mathbf{C}_M + \lambda \mathbf{I})^{-1} \mathbf{g}_0}$$

C’est la méthode qui a été utilisée pour la reconstruction de sources par beamformer.

Une autre solution, appelée “Eigenspace-Projected Beamformer”, consiste à projeter les poids du filtre sur le sous-espace contenant le signal (et

4. Utilisation de la reconstruction de l'activité corticale pour les ICM1

non le bruit) de la matrice de covariance des mesures [Van Veen, 1988, Sekihara *et al.*, 2002b]. Cette approche est similaire celle utilisée par la méthode MUSIC pour la localisation de dipôles [Mosher *et al.*, 1992]. Les sous-espaces du signal et du bruit sont déterminés par diagonalisation de \mathbf{C}_M , les valeurs propres les plus petites étant supposées être dues au bruit :

$$\mathbf{C}_M = \mathbf{E}_S \mathbf{D}_S \mathbf{E}_S^T + \mathbf{E}_N \mathbf{D}_N \mathbf{E}_N^T \quad (\text{V.6})$$

où $\mathbf{D}_S = \text{diag}[\lambda_1, \dots, \lambda_Q, 0, \dots, 0]$ et $\mathbf{D}_N = \text{diag}[0, \dots, 0, \lambda_{Q+1}, \dots, \lambda_{NE}]$ avec $\lambda_1 \geq \dots \geq \lambda_{NE}$.

On peut donc exprimer les poids du beamformer défini dans l'équation (V.5) par :

$$\begin{aligned} \mathbf{w}_{s_0} &= \frac{\mathbf{C}_M^{-1} \mathbf{g}_0}{\mathbf{g}_0^T \mathbf{C}_M^{-1} \mathbf{g}_0} \\ &= \frac{(\mathbf{E}_S \mathbf{D}_S \mathbf{E}_S^T + \mathbf{E}_N \mathbf{D}_N \mathbf{E}_N^T)^{-1} \mathbf{g}_0}{\mathbf{g}_0^T \mathbf{C}_M^{-1} \mathbf{g}_0} \\ &= \frac{\mathbf{E}_S \mathbf{D}_S^{-1} \mathbf{E}_S^T \mathbf{g}_0}{\mathbf{g}_0^T \mathbf{C}_M^{-1} \mathbf{g}_0} + \frac{\mathbf{E}_N \mathbf{D}_N^{-1} \mathbf{E}_N^T \mathbf{g}_0}{\mathbf{g}_0^T \mathbf{C}_M^{-1} \mathbf{g}_0} \end{aligned} \quad (\text{V.7})$$

Théoriquement, \mathbf{g}_0 fait partie de l'espace des sources ce qui le rend orthogonal à \mathbf{E}_N et donc $\mathbf{E}_N^T \mathbf{g}_0 = 0$. Le membre de droite de l'équation (V.7) doit donc être nul. En réalité, ce terme peut ne pas être nul ce qui réduit le rapport signal sur bruit [Sekihara *et al.*, 2002b]. Les poids du "Eigenspace-Projected Beamformer" sont donc défini uniquement par le terme de gauche de l'équation (V.7) :

$$\tilde{\mathbf{w}}_{s_0} = \frac{\mathbf{E}_S \mathbf{D}_S^{-1} \mathbf{E}_S^T \mathbf{g}_0}{\mathbf{g}_0^T \mathbf{C}_M^{-1} \mathbf{g}_0}$$

D'après [Feldman et Griffiths, 1991] les poids du "Eigenspace-Projected Beamformer" peuvent être calculés par projection dans l'espace du signal des poids du beamformer classique soit :

$$\tilde{\mathbf{w}}_{s_0} = \mathbf{E}_S \mathbf{E}_S^T \mathbf{w}_{s_0}$$

4 UTILISATION DE LA RECONSTRUCTION DE L'ACTIVITÉ CORTICALE POUR LES ICM

4.1 Modification de la chaîne de traitement d'une ICM

L'intégration de la reconstruction de l'activité cérébrale dans la chaîne de traitement d'une ICM est théoriquement facile : une fois le prétraitement des données

effectué, on reconstruit l'activité corticale, puis on utilise le signal de chaque source comme s'il s'agissait du signal provenant d'un capteur. Cependant, plusieurs problèmes peuvent se poser :

- Reconstruire fidèlement l'activité corticale en temps réel est une tâche difficile.
- Le nombre de sources est bien supérieur au nombre de capteurs. Certains calculs, rapides sur les capteurs, peuvent être difficiles à réaliser en temps réel sur les sources¹⁰.
- Certains traitements sur les capteurs n'ont plus de sens sur les sources, en particulier les filtrages spatiaux comme le laplacien. Il s'agit alors de définir de nouveaux traitements basés sur les caractéristiques des sources.

L'utilisation de la reconstruction de l'activité corticale n'est donc qu'une première étape pour l'amélioration des ICM. Pour réellement augmenter les performances des ICM, des algorithmes spécifiques doivent être conçus pour tirer parti de l'information présente dans les sources. Il s'agit par exemple d'utiliser la nature physiologique des sources, ou encore de tirer parti de la position des différentes sources.

4.2 Sélection de features

Le nombre de sources étant bien plus élevé que le nombre de capteurs, reconstruire l'activité corticale multiplie le nombre de features utilisables.

D'une certaine façon, plus la dimension de l'espace des features est important, c'est à dire que plus l'on a de features, plus l'on dispose d'information sur chaque essai ce qui améliore potentiellement la classification. Cependant, toutes les features ne contiennent pas d'information utile à la classification et plus la dimension de l'espace des features est grand plus le risque de sur-apprentissage est important. En effet, quand la dimension de l'espace des features augmente, le nombre de données d'apprentissage nécessaires pour permettre une bonne généralisation explose. Ce phénomène est connu sous le nom de "Malédiction de la dimension" ou "*Curse of dimensionality*". Une fois l'activité corticale reconstruite il est donc quasiment indispensable d'utiliser une méthode de sélection de features avant de pouvoir effectuer un apprentissage [Besserve *et al.*, 2008].

Deux approches existent pour sélectionner les features [Blum et Langley, 1997] :

- l'approche de type filtrage.
- l'approche par apprentissage successif dite "*wrapper method*".

L'approche de type filtrage consiste à sélectionner les features les plus utiles (ou éliminer les moins informatives) avant d'effectuer un apprentissage. Au contraire,

¹⁰Le beamformer et le Minimum-Norm étant des méthodes linéaires, certains calculs peuvent être faits avant la reconstruction des sources pour réduire les temps de calculs.

4. Utilisation de la reconstruction de l'activité corticale pour les ICM3

les méthodes par apprentissage successif effectuent des apprentissages et des tests en utilisant différents sous-ensembles de features afin de déterminer le sous-ensemble optimal des features à utiliser. L'approche par apprentissage successif permet d'obtenir de meilleures performances [Kohavi et John, 1997], mais engendre des temps de calcul bien plus importants dû au nombre d'apprentissages et de tests effectués. Le nombre de features pouvant être extrêmement élevé une fois l'activité corticale reconstruite, il n'est pas possible d'utiliser des méthodes de type *wrapper*.

Les méthodes de type filtrage peuvent être séparées en deux catégories suivant comment sont évaluées les performances des features :

- Évaluation indépendante de chaque feature. L'utilité de chaque feature est évaluée indépendamment en utilisant par exemple le t-test de Welch.
- Évaluation globale. Il s'agit d'évaluer de façon globale la contribution de chaque feature à la classification. La méthode LASSO est ainsi souvent utilisée pour effectuer de la sélection de features [Tibshirani, 1996].

L'inconvénient majeur des méthodes utilisant une évaluation indépendante des features est qu'elles ne prennent pas en compte la redondance de l'information entre les différentes features. Ceci est particulièrement gênant dans le cas de features extraites à partir de l'activité corticale : le nombre de sources étant plus important que le nombre de capteurs, les sources voisines sur le cortex sont extrêmement corrélées. Une méthode de filtrage utilisant une évaluation indépendante des features risque de sélectionner l'ensemble des features provenant d'une même région corticale, alors qu'un sous-ensemble aurait été suffisant. Inversement, des features légèrement moins performantes provenant d'une autre région mais apportant une information complémentaire risquent de ne pas être sélectionnées.

Pour éviter ce phénomène nous avons utilisé un algorithme récursif, basé sur le t-test de Welch, sélectionnant à chaque étape la feature ajoutant le plus d'information aux features déjà sélectionnées. Les détails de cet algorithme sont donnés dans [Fruitet *et al.*, 2010].

4.3 Coût de l'utilisation de la reconstruction de sources

La reconstruction de l'activité corticale rend plus complexes le développement et l'utilisation d'ICM.

Rallongement du temps d'installation du casque EEG Afin de reconstruire l'activité corticale, un grand nombre d'électrodes doivent être enregistrées. L'installation d'une dizaine d'électrodes prend environ 5 minutes et il faut donc compter entre 30 minutes et une heure pour installer 64 ou 128 électrodes. De plus, la localisation des électrodes doit être enregistrée. Cette localisation se fait généralement à l'aide d'un dispositif de pointage 3D et doit être effectuée pour chaque électrode. Cela prend environ 10 minutes quand tout fonctionne parfaitement.

Création du modèle de tête Si l'on veut utiliser un modèle de tête réaliste, le sujet doit passer une IRM anatomique, puis un modèle de tête doit être généré à partir des images segmentées. Les méthodes entièrement automatiques ne donnant que rarement de bons résultats, il est généralement nécessaire d'ajuster un grand nombre de paramètres manuellement.

La reconstruction de sources est donc un processus coûteux en données et en temps, qui ne sera utile aux ICM que s'il permet un gain important au niveau des performances.

5 RÉSULTATS

5.1 Comparaison des méthodes sur des données d'imagination motrice

Afin de comparer les performances obtenues en reconstruisant l'activité corticale par Minimum-Norm et beamformer, nous avons utilisé les enregistrements décrits dans la première partie du chapitre III. Il s'agit d'enregistrements d'imagination de mouvements durant des périodes de 8 secondes. Pour le sujet A1 six mouvements différents ont été imaginés et seulement trois (main droite, gauche et les pieds) pour le sujet A2. Chaque période d'imagination de 8 secondes est découpée en 29 fenêtres de 1 seconde que l'on va utiliser pour retrouver quel mouvement était imaginé. L'objectif étant de mesurer les taux de discrimination entre les différentes tâches, nous avons effectué une classification binaire entre chaque couple de tâches.

Le protocole d'acquisition et la chaîne de traitement, qui ici est modifiée par l'inclusion de la reconstruction de sources, ont été décrits dans la section 1 du chapitre III (page 69). La chaîne de traitement est ainsi composée de :

1. Pre-traitements : filtrage entre 4 et 40Hz ;
2. Reconstruction de l'activité des sources ;
3. Extraction des features (spectre de puissance) ;
4. Sélection des features discriminantes ;
5. Classification à l'aide d'un SVM linéaire.

5.1.1 Résultats sujet A1

Le sujet A1 ne disposant pas d'une IRM, un modèle de tête réaliste mais non sujet-spécifique a été utilisé. Le sujet ayant réalisé 6 tâches d'imagination différentes, relativement peu d'essais ont été enregistrés pour chaque tâche ce qui rend l'apprentissage difficile et la comparaison entre les diverses méthodes délicate. Le taux de classification binaire moyen entre toutes les paires de tâches

est donné dans le tableau V.1. Le taux de classification moyen relativement bas peut être expliqué par la présence de couples de tâches que nous n'avons pas réussi à différencier (comme pied droit vs gauche, ou index droit vs main droite). L'objectif de l'expérience était de voir si l'utilisation de la reconstruction de sources permettrait de différencier des tâches indistinguables l'une de l'autre en utilisant directement l'information provenant des capteurs. Tel ne fut hélas pas le cas.

Comme on peut s'en apercevoir à la lecture des résultats (tableau V.1), seul le beamformer permet un très léger gain par rapport à l'utilisation d'un filtre laplacien. Ceci peut cependant être dû au nombre insuffisant d'essais pour chaque classe.

Méthode	Taux de classification moyen
Électrodes brutes	59.9%
Laplacien	60.9%
Minimum-Norm pondéré	60.7%
Beamformer	62.0%

Tab. V.1 – Taux de classification binaire moyen entre toutes les paires de tâches pour le sujet A1.

5.1.2 Résultats sujet A2

Un modèle de tête réaliste adapté au sujet a été réalisé à partir d'une IRM (voir figure V.3). Le taux de classification binaire entre les différentes tâches est donné dans le tableau V.2. Le beamformer permet des résultats équivalents à l'utilisation d'un filtre laplacien. Seule la reconstruction de source par Minimum-Norm permet un très léger gain de performance.

Méthode	MD/MG	MD/P	MG/P	moyenne
Électrodes brutes	68%	75%	71%	70.9%
Laplacien	69%	81%	75%	75.1%
Minimum-Norm	76%	82%	72%	76.6%
Minimum-Norm pondéré	77%	81%	74%	77.2%
Beamformer	75%	75%	74%	74.7%

Tab. V.2 – Taux de classification binaire entre la main droite (MD) la gauche (MG) et les pieds (P) pour le sujet A2.

5.1.3 Localisation des sources pour le sujet A2

La figure V.4 représente la localisation des sources sur le cortex permettant l'extraction de features utiles à la classification. L'électrode C4 (au dessus de l'aire

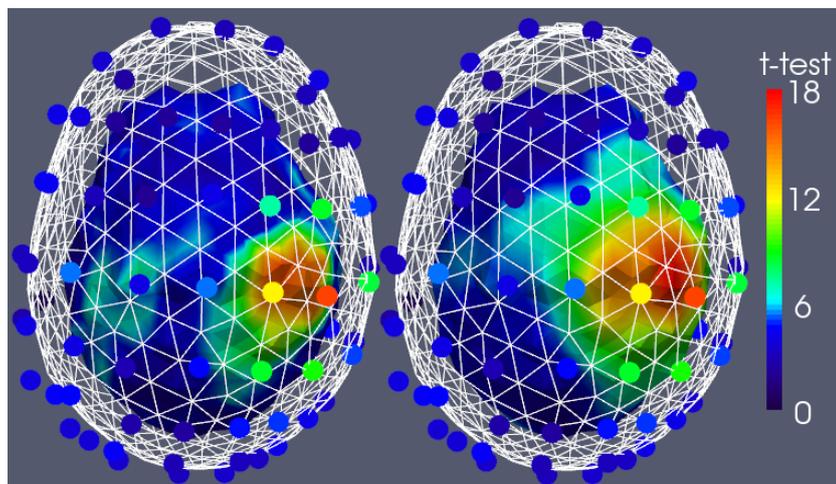


Fig. V.4 – Localisation des features discriminantes pour distinguer l'imagination d'un mouvement de la main droite d'un de la gauche pour le sujet A2. La reconstruction de l'activité corticale est effectuée par Minimum-Norm pondérée à gauche et par beamformer à droite. La localisation des électrodes avec les valeurs de t -test est représentée sur chaque figure.

motrice de la main gauche) est l'électrode ayant le plus haut t -test, l'électrode C3 n'apportant que peu d'information pour la classification. Le beamformer (à droite sur la figure V.4) permet d'obtenir une large population de sources avec un t -test élevé au niveau de l'aire motrice de la main gauche (donc sur l'hémisphère droit). L'activité utile après reconstruction par Minimum-Norm est plus focale et l'on distingue quelques sources ayant un t -test modéré sur l'aire motrice de la main droite.

5.2 Discussion

5.2.1 Difficultés pour l'interprétation des résultats

Les résultats présentés dans ce chapitre sur l'utilisation de la reconstruction de l'activité cérébrale sont difficilement interprétables. Bien qu'un très léger gain ait pu être obtenu grâce à l'utilisation du beamformer pour le sujet A1 et du Minimum-norme pour le sujet A2, le nombre de sujets et d'expériences n'est pas suffisant pour valider et quantifier le gain lié à l'utilisation de la reconstruction de sources.

D'autre part, l'utilisation de la reconstruction de sources pour les ICM étant particulièrement coûteuse, un gain aussi réduit (entre 1 et 2% par rapport à la simple utilisation d'un filtre laplacien) n'est pas suffisant pour justifier un tel surcoût.

Cependant, comme expliqué dans le début de ce chapitre, la reconstruction de

l'activité corticale n'est qu'une étape permettant éventuellement une amélioration du fonctionnement des ICM. Si la reconstruction de sources permet un gain de performances, il est fort probable que ce soit grâce à l'utilisation d'algorithmes pour l'extraction des features et la classification tirant parti de la nouvelle nature du signal. Plus précisément, il s'agit de développer de nouvelles features et de nouveaux traitements utilisant l'information de localisation ou des informations physiologiques des sources corticales. On citera par exemple le travail de Michel Besserve qui exploite des mesures de connectivité entre les sources corticales [Besserve, 2007, Besserve *et al.*, 2011].

5.2.2 Méta analyse de la reconstruction de l'activité cérébrale pour les ICM

Un état de l'art sur l'utilisation de la reconstruction de l'activité cérébrale pour les ICM est réalisé dans la section 3.3.2 du chapitre II (page 62). Pour l'établir nous avons analysé une quinzaine de publications portant sur la reconstruction de sources pour les ICM¹¹.

La totalité de ces publications rapporte un gain de performances obtenu grâce à la reconstruction de l'activité cérébrale. Cependant, l'ensemble des publications que nous avons analysées portent sur des méthodes de reconstruction de sources développées (ou adaptées) par les auteurs ce qui pose des problèmes d'objectivité¹². D'autre part, les auteurs comparent généralement leurs méthodes à des utilisations directes des mesures aux capteurs et non aux performances qui pourraient être obtenues avec des méthodes plus complexes. Des comparaisons plus objectives, entre l'utilisation de la reconstruction des sources et les dernières méthodes de l'état de l'art de la recherche sur les ICM, sont donc nécessaires pour juger de l'utilité réelle de la reconstruction de l'activité cérébrale.

Dans [Ahn *et al.*, 2010], les auteurs comparent l'utilisation de la reconstruction de sources au CSP. Cependant, le gain de performances miraculeux obtenu sur 2 des 5 sujets laisse penser que le CSP sur les électrodes brutes était mal paramétré et a donné lieu à un sur-apprentissage. Dans [Grosse-Wentrup *et al.*, 2009], les auteurs rapportent que le beamformer permet de meilleures performances que le CSP quand de nombreux artefacts sont présents.

Plusieurs méthodes de reconstruction de sources ont été appliquées sur des jeux de données publiques et en particulier sur ceux des "BCI compétition". Dans [Ahn *et al.*, 2012] les performances obtenues grâce à la reconstruction de l'ac-

¹¹[Qin *et al.*, 2004, Kamousi *et al.*, 2005, Grave de Peralta Menendez *et al.*, 2005, Congedo *et al.*, 2006, Besserve, 2007, Babiloni *et al.*, 2007, Lotte *et al.*, 2008a, Lotte, 2008, Noirhomme *et al.*, 2008, Grosse-Wentrup *et al.*, 2009, Ahn *et al.*, 2010, Besserve *et al.*, 2011, Ahn *et al.*, 2012]

¹²Il ne s'agit pas de remettre en question l'honnêteté des auteurs, mais simplement, à notre avis, des analyses extérieures objectives sont nécessaires pour conclure à l'utilité réelle de la reconstruction de sources pour les ICM.

tivité cérébrale sont bien inférieures à celles obtenues durant la compétition alors que dans [Lotte *et al.*, 2008a] un gain de 1 à 2% est obtenu.

En l'état actuel des connaissances, nous pensons qu'il n'est pas possible de déterminer si la reconstruction de l'activité cérébrale permet d'améliorer les performances des ICM.

5.2.3 Vers une "BCI competition" permettant la reconstruction des sources

L'utilisation des méthodes de reconstruction de l'activité cérébrale étant particulièrement complexe et nécessitant une certaine expertise, il est difficile de réaliser une comparaison objective extérieure de ces méthodes. De notre point de vue, une façon d'analyser objectivement et finement le gain permis par l'utilisation de la reconstruction de l'activité cérébrale est d'organiser une nouvelle "BCI competition" ayant pour but de comparer les performances d'ICM traditionnelles à celles utilisant la reconstruction de l'activité cérébrale.

Afin de permettre une comparaison fiable, nous pensons que cette compétition devrait respecter les points suivants :

1. Le nombre de sujets sur lesquels porte la compétition doit être important (5 minimum).
2. Pour chaque sujet, les enregistrements doivent être effectués sur plusieurs jours pour vérifier la robustesse inter-sessions. Deux sessions pourraient servir pour effectuer l'apprentissage et une troisième pour l'évaluation des performances¹³.
3. Les soumissions doivent être effectuées dans trois catégories :
 - (a) Sans reconstruction de sources. Seuls les enregistrements des mesures au niveau des capteurs sont utilisés.
 - (b) Reconstruction de sources approchée. On utilise la localisation spatiale des électrodes, mais l'on ne dispose pas d'un modèle anatomique adapté aux sujets.
 - (c) Reconstruction avec modèle adapté au sujet. On utilise la localisation des électrodes ainsi que des modèles de têtes réalistes basés sur les IRM des sujets.
4. Pour permettre au maximum de concurrents de participer à la reconstruction de sources, des modèles de têtes ainsi que des matrices de gain pourraient être fournis.

¹³Les enregistrements des trois sessions seraient fournis ainsi que les labels des 2 premières sessions mais pas de la troisième.

6 CONCLUSION

Nous avons étudié dans ce chapitre la possibilité de reconstruire l'activité corticale en temps réel pour augmenter la précision spatiale de l'EEG et ainsi améliorer les performances des ICM. Nous nous sommes intéressés à deux types de méthodes en particulier, à savoir la reconstruction par "Minimum-Norm" et par filtrage de type "beamformer". Nous avons utilisé des modèles de têtes réalistes, dont certains basés sur des images IRM des sujets. Bien que nous ayons réussi à utiliser ces méthodes en temps réel, ni cette étude ni la méta-analyse bibliographique réalisée n'ont permis de conclure que la reconstruction de l'activité corticale permette réellement d'augmenter les performances des ICM.

Toolbox Matlab pour les ICM

LES résultats présentés dans les chapitres III, IV et V ont été obtenus grâce à une boîte à outils Matlab que nous avons développée : *MoffBCI*¹. L'objectif de cette boîte à outil est de permettre une sélection automatique des méthodes les plus appropriées au traitement de chaque ICM, voire de chaque sujet, ainsi qu'un ajustement automatique de l'ensemble des paramètres utilisés. Il est ainsi possible de sélectionner automatiquement les prétraitements, les algorithmes d'extraction de features ou encore de classification utilisée. Par une analyse hors-ligne poussée *MoffBCI* permet ainsi une adaptation optimale à chaque sujet. D'autre part, la sélection automatique des paramètres permet de comparer honnêtement différentes méthodes, en s'assurant que pour chaque méthode les paramètres sont bien ajustés.

MoffBCI a été conçue pour être facilement réutilisable. Son fonctionnement, ainsi que les diverses méthodes disponibles pour l'évaluation des performances, est décrit dans ce chapitre. L'annexe A donne un exemple simple d'utilisation et l'annexe B est consacré à l'implémentation d'une fonction de la boîte à outils.

¹*MoffBCI* est l'acronyme de Matlab off-line Brain Computer Interface.

Sommaire

1	<i>MoffBCI</i>, une nouvelle toolbox pour les ICM	163
1.1	Pourquoi une nouvelle toolbox pour les ICM?	163
1.1.1	<i>Les logiciels permettant de développer une ICM</i>	<i>163</i>
1.1.2	<i>Quelques mots sur OpenViBE</i>	<i>163</i>
1.1.3	<i>MoffBCI, un complément d'OpenViBE pour l'analyse hors-ligne</i>	<i>163</i>
1.2	Cahier des charges de <i>MoffBCI</i>	165
1.2.1	<i>Une toolbox ouverte et modulaire</i>	<i>165</i>
1.2.2	<i>Au centre de MoffBCI : la comparaison de méthodes et de paramètres</i>	<i>165</i>
2	 Fonctionnement de <i>MoffBCI</i>	167
2.1	Les fonctions de traitement	167
2.2	La définition d'une chaîne de traitement	167
2.2.1	<i>Enregistrement des fonctions de traitement à utiliser</i>	<i>167</i>
2.2.2	<i>La spécification des paramètres</i>	<i>168</i>
2.3	Quelques détails sur les propriétés du noyau	168
2.3.1	<i>Calcul en arborescence</i>	<i>168</i>
2.3.2	<i>Parallélisation des calculs</i>	<i>169</i>
3	 Méthodes d'évaluation en apprentissage	171
3.1	Quelques notions sur l'apprentissage supervisé	171
3.1.1	<i>Définitions et notations</i>	<i>171</i>
3.1.2	<i>Évaluation de l'apprentissage supervisé</i>	<i>172</i>
3.2	Les méthodes d'évaluation de <i>MoffBCI</i>	173
3.2.1	<i>L'évaluation standard d'une chaîne de traitement (ou hold-out method)</i>	<i>173</i>
3.2.2	<i>La validation croisée</i>	<i>174</i>
3.2.3	<i>Évaluation tripartite ou 3-way data split</i>	<i>176</i>
3.2.4	<i>La double validation croisée</i>	<i>177</i>

1 *MoffBCI*, UNE NOUVELLE TOOLBOX POUR LES ICM

1.1 Pourquoi une nouvelle toolbox pour les ICM ?

1.1.1 Les logiciels permettant de développer une ICM

Il existe deux possibilités pour réaliser des expérimentations sur les ICM. On peut soit développer ses propres programmes, en partant des signaux enregistrés pour arriver jusqu'au signal de contrôle en passant par la gestion des stimulations et des feedbacks, soit utiliser des logiciels spécialement développés à ces fins.

Le logiciel BCI2000² est le plus connu, mais il en existe d'autres comme xBCI [Susila *et al.*, 2010] ou encore le plugin BCILAB [Delorme *et al.*, 2011] de la toolbox EEGLAB. Un bref descriptif des principaux logiciels utilisés pour développer des ICM est réalisé dans [Brunner *et al.*, 2011]

Le choix a été fait au sein de l'équipe Athéna d'utiliser OpenViBE [Renard *et al.*, 2010] développé initialement à l'INRIA Rennes et l'INSERM pour réaliser les expérimentations EEG.

1.1.2 Quelques mots sur OpenViBE

Une ICM est constituée dans le logiciel OpenViBE de "boîtes" reliées entre elles par des flux. Chaque boîte s'occupe d'une étape de traitement, comme par exemple l'application d'un filtre sur le signal EEG ou l'affichage d'un stimulus à l'écran.

Pour l'instant le nombre de boîtes disponibles est limité, mais avec une bonne maîtrise du C++ il est possible de réaliser ses propres boîtes. C'est d'ailleurs, une des contributions de cette thèse, que d'avoir créé ou amélioré des boîtes OpenViBE.

OpenViBE est par définition très ouvert et modulaire, mais a été conçu pour faire fonctionner en-ligne des ICM et non pour évaluer ou optimiser les performances par des analyses hors-ligne. Son utilisation est ainsi peu adaptée à l'optimisation des paramètres ou à la comparaison hors-ligne de différentes techniques.

1.1.3 *MoffBCI*, un complément d'OpenViBE pour l'analyse hors-ligne

L'idée originelle de *MoffBCI* était de créer une toolbox totalement axée sur l'évaluation des méthodes et sur l'optimisation automatique des paramètres et ainsi offrir un complément à OpenViBE. À notre connaissance, il n'existe pas d'autre boîte à outils ou librairie dédiées aux ICM permettant une comparaison totalement automatique des méthodes et des paramètres.

Le but de *MoffBCI* est de pouvoir rapidement implémenter de nouvelles méthodes de traitement, d'évaluer et de comparer leurs performances par des analyses hors-ligne, puis de transférer dans OpenViBE les plus adaptées. Pour permettre

²www.bci2000.org

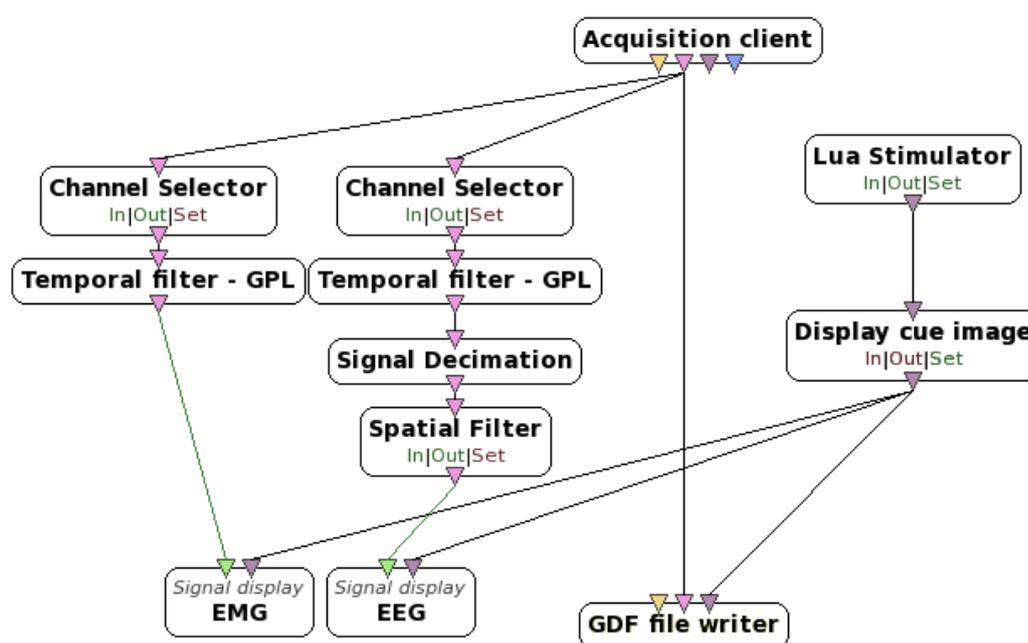


Fig. VI.1 – Exemple de scénario OpenViBE. La partie de gauche permet de filtrer et d’afficher les signaux EEG et EMG. La partie de droite permet d’afficher des images sur l’écran de stimulation. Le signal brut ainsi que les instants où les images sont affichées sont enregistrés dans un fichier gdf pour un traitement hors-ligne

un développement rapide le choix du langage de programmation s'est porté sur Matlab et non sur le C++ (qui est le langage d'OpenViBE). La transition de fonctions depuis *MoffBCI* vers OpenViBE est facilitée par une boîte (à laquelle j'ai contribué) permettant d'utiliser du code Matlab dans OpenViBE.

1.2 Cahier des charges de *MoffBCI*

1.2.1 Une toolbox ouverte et modulaire

La plupart des boîtes à outils ou bibliothèques dédiées aux ICM divisent le fonctionnement des ICM en étapes cadrées et obligatoires :

- Le signal biologique doit être fourni dans une structure de données précise.
- On applique éventuellement des pré-traitements.
- On extrait les features contenant l'information intéressante.
- On convertit les features en résultats à l'aide d'algorithmes d'apprentissage supervisé. Cette dernière étape peut être englobée dans une méthode de validation (comme la validation croisée).

Pour *MoffBCI* il n'y a pas de schéma type. Suivant le même principe qu'OpenViBE, la toolbox est constituée de deux parties :

- **un noyau** établissant un lien entre les traitements et fournissant des méthodes d'évaluation et d'ajustement des paramètres.
- **un ensemble de fonctions** permettant divers traitements que l'utilisateur peut facilement enrichir ou utiliser indépendamment du reste de la boîte à outils.

Un prototype d'ICM est alors défini par un enchaînement de fonctions de traitement. Les sorties des premières fonctions appelées constituent les entrées des fonctions suivantes.

Le schéma VI.2 donne un exemple de traitement réalisé avec *MoffBCI*.

1.2.2 Au centre de *MoffBCI* : la comparaison de méthodes et de paramètres

La base du noyau est de lier les sorties des premières fonctions appelées aux entrées des suivantes, permettant ainsi de faire fonctionner l'ensemble de la chaîne de traitement. Cependant des fonctionnalités plus poussées permettent pour chaque fonction de traitement d'essayer divers paramètres ou diverses méthodes. La meilleure combinaison de méthodes avec des paramètres ajustés peut ainsi être déterminée par des méthodes d'évaluation comme la validation croisée [Kohavi, 1995]. De même, diverses méthodes de traitement peuvent être évaluées tout en garantissant l'utilisation de paramètres optimaux pour chacune d'entre elles.

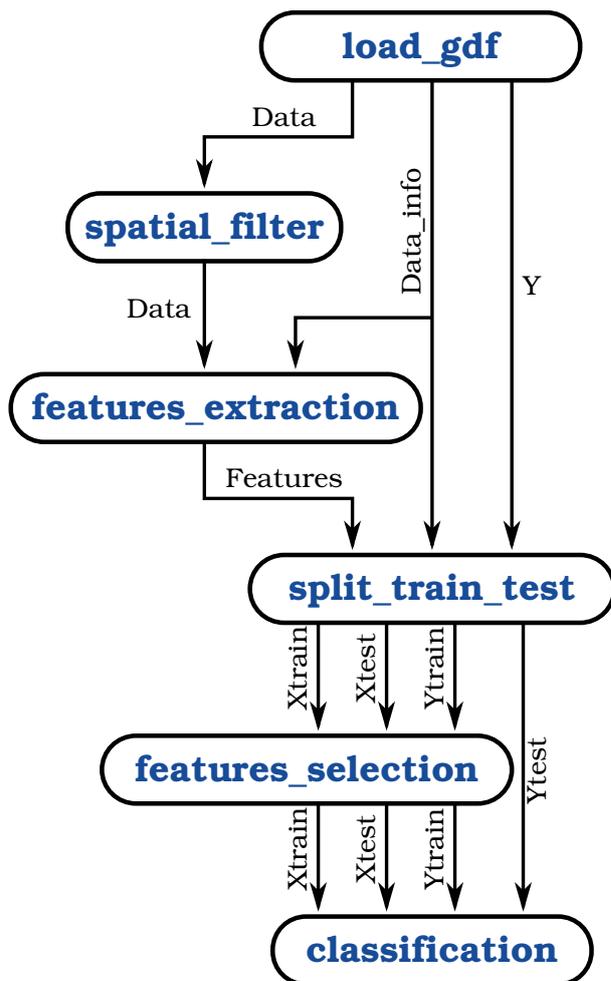


Fig. VI.2 – Exemple de chaîne de traitement réalisée avec *MoffBCI*.

2 FONCTIONNEMENT DE *MoffBCI*

2.1 Les fonctions de traitement

Les fonctions de traitement de *MoffBCI* sont l'équivalent des boîtes dans Open-ViBE. Ce sont les éléments de base qui effectuent les traitements élémentaires, comme l'application d'un filtre laplacien ou l'apprentissage et l'évaluation d'un SVM. Ce sont des fonctions Matlab standard, la seule contrainte étant que leur premier argument soit une structure contenant les paramètres de la fonction.

$$[Output_1, Output_2\dots] = FonctionName(para, Input_1, Input_2\dots)$$

L'écriture de nouvelles fonctions est donc extrêmement simple et flexible (un exemple de fonction est donné dans l'Annexe B). Un des objectifs de *MoffBCI* est de permettre de tester de nouvelles méthodes et algorithmes aussi facilement que possible.

2.2 La définition d'une chaîne de traitement

La réalisation d'une chaîne de traitement avec *MoffBCI* se fait par le remplissage d'une structure, que nous appellerons dans notre exemple *ProcessStruct*. Ceci est effectué en deux étapes détaillées ci-dessous.

2.2.1 Enregistrement des fonctions de traitement à utiliser

La première étape consiste à déclarer les fonctions de traitement utilisées. Cela se fait en remplissant le champ *functions* de la structure *ProcessStruct*. Pour chaque fonction il suffit de fournir :

- le nom de la fonction (champ *name*)
- le nom des variables qu'elle prend en argument en sus de la structure de paramètres (champ *global_in*)
- les nom des variables de sortie (champ *global_out*)

D'autres champs sont disponibles pour permettre plus de flexibilité comme le chargement ou l'enregistrement de variables avant ou après l'appel à la fonction, ou encore permettre des instances multiples d'une même fonction.

Les deux premières fonctions de la chaîne de traitement représentée sur le schéma VI.2 seraient implémentées comme suit :

```

1 ProcessStruct.functions(1).name      = 'load_gdf';
2 ProcessStruct.functions(1).global_in = {};
3 ProcessStruct.functions(1).global_out = {'Data', 'Y', 'Data_info'};
4
5 ProcessStruct.functions(2).name      = 'spatial_filter';
6 ProcessStruct.functions(2).global_in = {'Data'};
7 ProcessStruct.functions(2).global_out = {'Data'};
```

2.2.2 La spécification des paramètres

La spécification des paramètres se fait par simple ajout à la structure de champ comportant les noms des paramètres.

Prenons l'exemple de la fonction *spatial_filter*, qui permet de réaliser un filtrage laplacien spatial. Elle comporte un unique paramètre, appelé *spatial_filter*, qui permet de déterminer la taille du filtre. On utilisera un filtre laplacien "large" grâce à l'ajout du champ :

```
1 ProcessStruct.spatial_filter = 'large';
```

Maintenant si l'on désire essayer différentes valeurs pour un paramètre, il suffit de remplacer la valeur du paramètre par une structure comportant deux champs :

- le nom de la fonction qui utilise ce paramètre (ainsi le noyau sait avant l'appel de quelle fonction il doit essayer les différentes valeurs du paramètre),
- la liste des valeurs à essayer.

Cela donnerait pour notre exemple :

```
1 ProcessStruct.spatial_filter.function = 'spatial_filter';
2 ProcessStruct.spatial_filter.values = { 'none', 'small', 'large' };
```

L'intérêt de *MoffBCI* réside dans le fait qu'il est très facile de faire varier en même temps un grand nombre de paramètres. On peut ainsi déterminer les valeurs les plus appropriées pour l'ensemble des paramètres ou encore mesurer l'impact de chaque paramètre sur les résultats.

2.3 Quelques détails sur les propriétés du noyau

2.3.1 Calcul en arborescence

Bien que la chaîne de traitement soit définie comme une liste de fonctions, son évaluation dans *MoffBCI* se déploie sous la forme d'un arbre : chaque fonction de la chaîne est un nœud de l'arbre qui a autant de branches qu'il y a de valeurs différentes pour ses paramètres.

Gestion des variables Un travail d'optimisation a été effectué au niveau de la gestion des variables. Pour éviter de calculer la totalité de la chaîne de traitement pour chaque valeur de paramètres, les variables d'entrée des fonctions de traitement sont temporairement gardées en mémoire quand cela est nécessaire.

Exemple

Dans l'exemple de la figure VI.2, l'entrée et la sortie de spatial_filter est Data. Si plusieurs filtres sont essayés, il faut sauvegarder l'état de Data avant l'application du 1^{er} filtre pour calculer le second.

D'autre part, le passage par référence ou par nom n'étant pas possible dans Matlab, des variables globales sont utilisées afin de minimiser leur recopie. Ainsi les

variables d'entrée et de sortie de toutes les fonctions de traitement sont déclarées globales par le noyau.

Gestion des résultats Les résultats sont enregistrés dans une structure en arbre où chaque nœud correspond à un paramètre. La figure VI.3 est une représentation d'un arbre de résultats.

Un ensemble de fonctions permet de manipuler les arbres de résultats de *MoffBCI*. Il est ainsi possible de récupérer les valeurs des paramètres donnant les meilleurs résultats, mais aussi de fusionner des arbres ou au contraire d'extraire un sous-arbre suivant la valeur d'un paramètre.

2.3.2 Parallélisation des calculs

L'évaluation de toutes les combinaisons de paramètres étant particulièrement coûteuse en temps de calcul, *MoffBCI* permet de distribuer les calculs de façon automatique sur plusieurs ordinateurs. Il suffit pour cela de lancer plusieurs processus Matlab qui communiqueront par TCP/IP :

- un processus sert de serveur et distribue les calculs,
- un processus récupère et stocke les résultats,
- un nombre arbitraire de processus servent de calculateurs.

Fonctionnement Avant d'évaluer une fonction pour laquelle différentes valeurs d'un paramètre doivent être essayées, un appel est effectué vers le serveur pour vérifier si des calculateurs sont libres. Si tel est le cas, les branches des calcul correspondant aux premières valeurs du paramètre sont transférées sur les calculateurs libres. Sinon le calcul continue séquentiellement.

Échange des données Afin de limiter les transferts de données entre les différents processus, à chaque variable est attribué un identifiant et une version permettant que seules les données nécessaires et non présentes sur les calculateurs soient échangées.

Sur les expériences effectuées, le temps de calcul engendré par les communications et les fonctions propres du noyau était négligeable par rapport au temps passé dans les fonctions de traitement.

Perspectives L'INRIA de Sophia Antipolis dispose d'une large ferme de calcul. Il serait intéressant de voir comment *MoffBCI* peut être adapté pour l'utiliser.

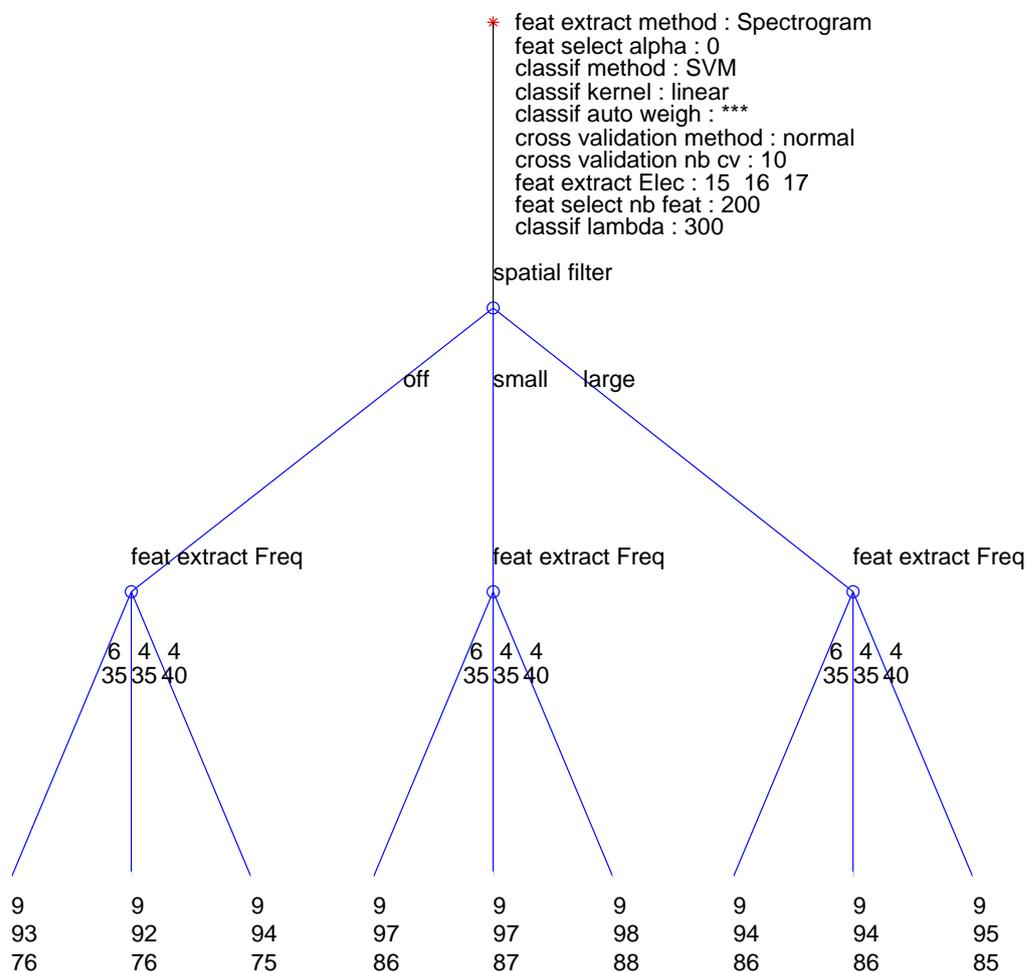


Fig. VI.3 – Arbre de résultats après évaluation d’une chaîne de traitement par validation croisée. Les paramètres affichés en haut sont fixés. Les autres constituent les nœuds de l’arbre. En pratique, le nombre de paramètres ajustés automatiquement est plus important, mais l’arbre de résultats est alors trop complexe pour être imprimé. Trois nombres sont représentés sous chaque feuille : le premier est le nombre de résultats qui sont regroupés dans cette feuille (pour la validation croisée il doit être identique pour chaque feuille et correspond au nombre d’ensembles). Le deuxième donne les performances en pourcentage de bien classés sur les données d’apprentissage et le troisième sur les données de test. Sur cet exemple, on constate que les meilleurs résultats sont obtenus avec un filtrage spatial de type laplacien “small”, par contre la bande de fréquence utilisée influe peu sur les performances.

3 MÉTHODES D'ÉVALUATION EN APPRENTISSAGE

3.1 Quelques notions sur l'apprentissage supervisé

3.1.1 Définitions et notations

L'apprentissage supervisé est une technique d'apprentissage automatique. On cherche à apprendre une fonction $h : \mathcal{X} \rightarrow \mathcal{Y}$ à partir d'une base de données d'exemples $\{x_i, y_i\}_{1 \leq i \leq N} \in (\mathcal{X} \times \mathcal{Y})$, puis à utiliser cette fonction sur de nouvelles observations de l'espace d'entrée \mathcal{X} .

L'espace d'entrée \mathcal{X} est l'ensemble des variables descriptives du problème ou *features*. On a généralement $\mathcal{X} = \mathbb{R}^d$, avec d le nombre de features.

Nous nous restreindrons au cas de la classification, c'est à dire au cas où l'ensemble de sortie \mathcal{Y} est discret. Il s'agit alors d'associer à chaque exemple décrit par un vecteur de features $x \in \mathcal{X}$ un label (ou étiquette) $y \in \mathcal{Y}$. Pour la régression, l'ensemble de sortie \mathcal{Y} est continu avec généralement $\mathcal{Y} = \mathbb{R}$. On cherche alors une fonction $h : \mathcal{X} \rightarrow \mathbb{R}$.

Définition de la perte On modélise le problème de classification par une distribution de probabilité \mathcal{D} sur $(\mathcal{X} \times \mathcal{Y})$. La base de données fournie en exemple est supposée tirée de façon identique et indépendante suivant cette loi \mathcal{D} .

Afin d'évaluer les performances de la fonction de classification h , il faut définir une fonction de perte $l : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}^+$. Sur un exemple fourni (x_i, y_i) , la perte liée à la prédiction de $h(x_i)$ au lieu de y_i est mesurée par $l(h(x_i), y_i)$.

On définit la perte empirique du classifieur h sur les données d'exemple $\hat{L}(h)$ par :

$$\hat{L}(h) = \frac{1}{N} \sum_{i=1}^N l(h(x_i), y_i)$$

La perte théorique du classifieur h est définie par :

$$L(h) = \mathbb{E}_{(x,y) \sim \mathcal{D}} (l(h(x), y))$$

Classifieur idéal et classifieur empirique optimal Le classifieur idéal, noté h^* , est celui qui permettrait des performances idéales en minimisant la perte théorique. Il est donc défini par :

$$h^* = \arg \min_{h \in \mathcal{H}} (L(h)) = \arg \min_{h \in \mathcal{H}} (\mathbb{E}_{(x,y) \sim \mathcal{D}} (l(h(x), y)))$$

Il n'est cependant pas possible de calculer h^* et il faut donc se contenter d'une version empirique. Le classifieur empirique optimal \hat{h}^* est quant à lui obtenu par minimisation de la perte empirique :

$$\hat{h}^* = \arg \min_{h \in \mathcal{H}} (\hat{L}(h))$$

soit

$$\hat{h}^* = \arg \min_{h \in \mathcal{H}} \left(\frac{1}{N} \sum_{i=1}^N l(h(x_i), y_i) \right)$$

3.1.2 Évaluation de l'apprentissage supervisé

L'évaluation des performances se fait à l'aide de la fonction de perte sur un ensemble de données. On notera $\hat{L}_{\mathbf{X}}(h)$ la perte du classifieur h sur un ensemble de données \mathbf{X} , définie par :

$$\hat{L}_{\mathbf{X}}(h) = \frac{1}{\text{card}(\mathbf{X})} \sum_{(x,y) \in \mathbf{X}} l(h(x), y)$$

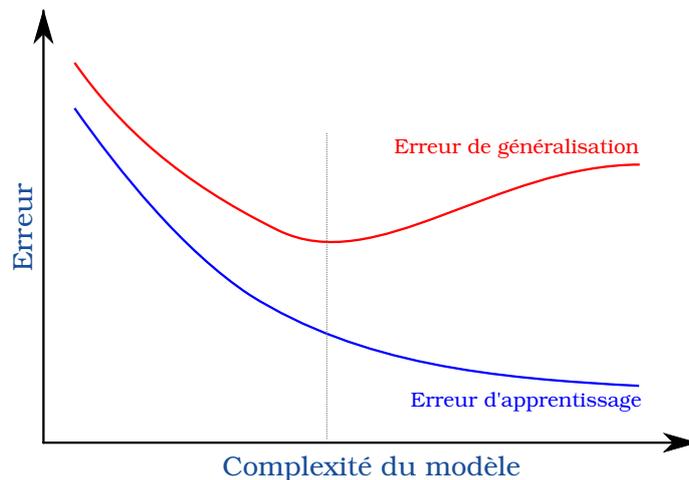


Fig. VI.4 – Évolution de l'erreur de classification en fonction de la complexité du modèle servant à construire le classifieur. Cette erreur est montrée sur l'ensemble d'apprentissage (bleu) et sur de nouvelles données (rouge).

Erreur d'apprentissage et erreur de généralisation Les erreurs de classification peuvent avoir deux origines.

- **Erreur d'apprentissage** Le classifieur n'est pas suffisamment complexe pour reconnaître les exemples fournis dans la base d'apprentissage.
- **Erreur de généralisation** Le classifieur ne parvient pas à généraliser l'information discriminante contenue dans la base d'apprentissage à de nouveaux essais.

La figure VI.4 représente l'évolution de l'erreur de classification sur la base d'apprentissage (erreur d'apprentissage) et sur de nouveaux exemples (erreur de généralisation). Plus le modèle est complexe, plus le classifieur est capable d'apprendre d'informations sur les exemples et donc plus il est performant sur la base

d'apprentissage. Cependant si le modèle est trop complexe, l'information enregistrée n'est pas pertinente vis à vis du problème et nuit à la généralisation. On parle alors de sur-apprentissage ou d'*overfitting*.

Le risque de sur-apprentissage est d'autant plus grand que la base de données servant pour l'apprentissage est réduite. En effet, si peu d'exemples sont donnés, il est difficile de différencier des variations liées au problème de variations aléatoires.

Dans le cas des ICM, les données devant être acquises par des expériences sur des sujets, les bases d'apprentissage sont extrêmement réduites. Le risque d'effectuer un sur-apprentissage est donc important et il vaut mieux se contenter de modèles peu complexes.

Séparation des données d'apprentissage et des données de test Afin d'évaluer les potentialités de généralisation d'un classifieur, il est extrêmement important de séparer les données utilisées pour l'apprentissage de celles utilisées pour évaluer les performances. Le problème est que ces deux ensembles doivent être de taille suffisante :

- si l'ensemble d'apprentissage est insuffisant, l'algorithme ne pourra pas généraliser à de nouveaux exemples et les performances seront donc médiocres ;
- si l'ensemble de test est trop réduit, l'évaluation des résultats sera imprécise.

3.2 Les méthodes d'évaluation de *MoffBCI*

Plusieurs méthodes d'évaluation sont disponibles dans *MoffBCI* suivant que l'on désire faire de la sélection de modèle ou évaluer les performances de diverses méthodes. L'annexe A donne un exemple d'utilisation de ces différentes méthodes.

3.2.1 L'évaluation standard d'une chaîne de traitement (ou holdout method)



Fig. VI.5 – Évaluation standard (ou holdout) : séparation des données en un ensemble d'apprentissage et un ensemble de tests disjoints.

L'inclusion de la fonction de traitement *split_train_test* dans la chaîne permet de séparer l'ensemble des données en deux sous-ensembles (voir figures VI.2 et VI.5) :

- un ensemble d'apprentissage,
- un ensemble de test.

La fonction de traitement *classification* permet d'utiliser différentes techniques de classification. L'apprentissage est effectué sur les données d'apprentissage et les performances sont évaluées sur les deux ensembles afin d'évaluer l'erreur

d'apprentissage et l'erreur de généralisation. Les résultats sont enregistrés dans un arbre où chaque nœud correspond à un paramètre et où les feuilles comportent les résultats sur les deux ensembles.

Attention

Afin de garantir la validité des résultats, il est important qu'aucune fonction de traitement (mis-à-part l'évaluation) n'utilise les labels de l'ensemble de test.

Inconvénients Cette méthode d'évaluation comporte deux problèmes majeurs :

1. Le nombre de données étant particulièrement limité dans les ICM, la séparation en deux ensembles réduit encore la taille des données d'apprentissage et des données de test. Cela risque d'engendrer des performances sous-optimales et surtout une évaluation approximative. Ceci peut être particulièrement gênant si l'on désire comparer des méthodes aux performances voisines.
2. L'évaluation se faisant en seulement deux temps (apprentissage, puis test), il n'est pas possible d'ajuster des paramètres automatiquement.

Supposons que l'on veuille choisir entre plusieurs valeurs d'un paramètre.

- Si l'on choisit la valeur donnant les meilleurs résultats sur l'ensemble d'apprentissage, on favorise le sur-apprentissage et le paramètre donnant les meilleurs résultats en généralisation ne sera pas sélectionné.
- Si l'on choisit la valeur donnant les meilleurs résultats sur l'ensemble de test, cela biaise les résultats, puisque l'ensemble de test est fait pour évaluer les performances finales et non pour ajuster les paramètres.

3.2.2 La validation croisée

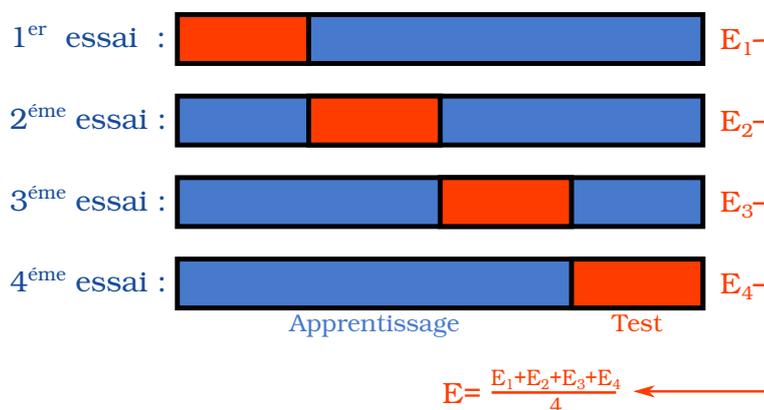


Fig. VI.6 – Validation croisée avec $n = 4$. Chaque essai est appelé fold, on parle alors de n-fold cross-validation

La technique de la validation croisée (voir figure VI.6) consiste à diviser les données en une partition de n ensembles et à utiliser alternativement chaque ensemble comme ensemble de test alors que l'apprentissage est réalisé sur le reste des données.

Si on note E_i l'estimateur de la performance sur l'ensemble de données i après avoir appris sur les ensembles $\{1, \dots, i-1, i+1, \dots, n\}$ l'estimateur de la performance globale est calculé par :

$$E = \frac{1}{n} \sum_{i=1}^n E_i$$

Implémentation dans *MoffBCI* Il est extrêmement simple de faire de la validation croisée avec *MoffBCI*. Il suffit de définir la partition des données à utiliser³ puis d'appeler une fonction du noyau pour évaluer la chaîne de traitement. Pour entrer un peu plus dans les détails, les ensembles d'apprentissage et de test sont passés comme des paramètres de la fonction *split_train_test*.

Les résultats sont, comme pour l'évaluation standard, enregistrés dans un arbre mais cette fois les feuilles comportent une liste de performances sur les données d'apprentissage et de test.

Avantages et inconvénients La validation croisée a plusieurs avantages :

- + Elle permet d'effectuer des apprentissages sur des ensembles plus grands. Les performances sont donc potentiellement meilleures.
- + L'ensemble des données est utilisé pour le test⁴ et l'évaluation est donc plus fiable et précise.
- + Plusieurs apprentissages et tests sont effectués. Il est donc possible d'effectuer des analyses statistiques sur les performances.

La validation croisée peut servir pour sélectionner la méthode et les paramètres les plus appropriés pour un ensemble de données, mais il n'est alors pas possible d'évaluer les performances finales sans biais. En effet il faudrait pour cela disposer de nouvelles données utilisées ni pour l'apprentissage, ni pour la sélection des paramètres.

Influence du nombre de sous-ensembles de la partition Quand le nombre de sous-ensembles n est grand :

- + le biais de l'estimateur de performance est réduit (l'estimation est précise)
- la variance de l'estimateur de performance est importante (les performances varient d'un ensemble de test à l'autre);
- le temps de calcul est important.

Quand le nombre de sous-ensembles n est faible :

³Une fonction de *MoffBCI* permet d'effectuer une partition des données en garantissant une répartition optimale des différentes classes dans les sous-ensembles.

⁴Pas en une seule fois, sinon ce serait une violation de la séparation des données d'apprentissage et de test

- le biais de l'estimateur de performance est plus important (mais cependant moins que pour une évaluation standard) ;
- + la variance de l'estimateur de performance est faible (les performances varient moins d'un ensemble à l'autre) ;
- + le temps de calcul est moins important.

3.2.3 Évaluation tripartite ou 3-way data split

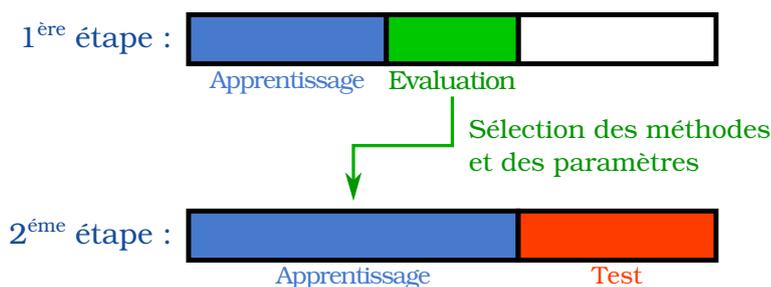


Fig. VI.7 – Évaluation tripartite ou 3-way data split. Les données sont séparées en trois ensembles : A pour l'apprentissage, E pour l'évaluation des paramètres et T pour le test.

L'évaluation tripartite est une amélioration de l'évaluation standard pour permettre de sélectionner automatiquement les paramètres les plus appropriés. Les données sont découpées en trois ensembles disjoints, que nous noterons A , E et T

1. Un premier apprentissage est réalisé sur l'ensemble A avec plusieurs méthodes et/ou différentes valeurs pour les paramètres.
2. On évalue les performances des méthodes et/ou des paramètres sur l'ensemble E et on sélectionne la méthode et/ou les paramètres donnant les meilleurs résultats.
3. On ré-effectue un apprentissage sur $A \cup E$
4. On teste les performances sur l'ensemble T .

implémentation dans *MoffBCI* La fonction *three_way_split* permet de réaliser simplement une évaluation tripartite. Il suffit de spécifier la partition des données en trois ensembles et les étapes sont effectuées de façon automatique. On obtient en résultats l'arbre des paramètres essayés sur l'ensemble E ainsi que les paramètres (ou méthodes) sélectionnés et les résultats finaux sur l'ensemble T .

Avantage et Inconvénients L'évaluation tripartite permet de sélectionner de façon automatique la méthode et les paramètres les plus adaptés aux données puis d'évaluer les performances sans introduire de biais. Cependant comme pour l'évaluation standard, la séparation des données en trois ensembles réduit et l'échantillon disponible pour l'apprentissage et celui pour le test.

3.2.4 La double validation croisée

La double validation croisée est une méthode de validation qui imbrique deux étapes de validation croisée afin de sélectionner automatiquement les méthodes et les paramètres les plus adaptés aux données et d'évaluer sans biais les résultats sur un nouvel ensemble de données.

Le fonctionnement de la double validation croisée est expliqué sur la figure VI.8. On effectue une partition en n sous-ensembles des données.

- À tour de rôle chacun des ensembles est mis de côté pour l'évaluation finale.
- Les $n - 1$ ensembles restants sont utilisés pour faire une nouvelle étape de validation croisée afin de sélectionner les meilleures méthodes et paramètres.
- Un nouvel apprentissage est effectué sur les $n - 1$ ensembles avec les meilleurs méthodes et paramètres.
- L'ensemble mis de côté est utilisé pour l'évaluation.

Avantages La double validation croisée permet :

- de sélectionner automatiquement les méthodes et paramètres adaptés en fonction des données
- d'effectuer une estimation sans biais des performances
- de maximiser la taille des ensembles sur lesquels sont effectués l'apprentissage, ce qui permet de meilleures performances.
- d'utiliser l'ensemble des données pour la validation, ce qui permet une estimation plus fiable et précise.

D'autre part, les méthodes et paramètres étant sélectionnés de façon indépendante pour chaque sous-ensemble de données, cela permet de vérifier si une méthode est systématiquement considérée comme plus performante ou si cela n'est pas significatif.

Inconvénients La double validation croisée engendre une multiplication des calculs nécessaires. Il est donc très important d'effectuer autant de calculs que possibles avant de séparer les données en ensembles d'apprentissage et de test. Ainsi, ces calculs ne seront effectués qu'une fois et non multipliés pour chaque séparation des données.

Utilisation dans *MoffBCI* L'utilisation de la double validation croisée dans *MoffBCI* est extrêmement simple. Comme pour la validation croisée simple, il suffit de définir la partition des données à utiliser puis d'appeler une fonction du noyau pour évaluer la chaîne de traitement.

Un exemple d'utilisation est donné dans l'annexe A pour chacune des méthodes d'évaluation.

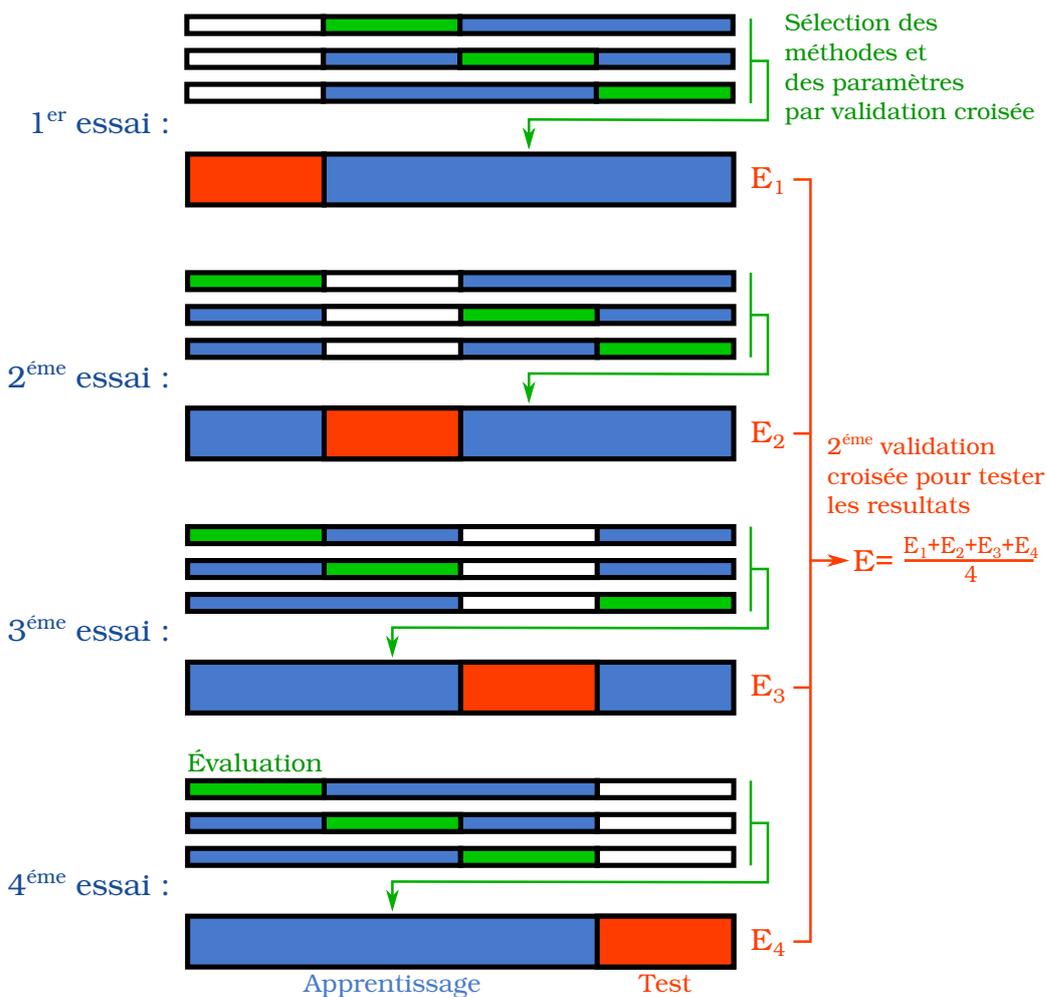


Fig. VI.8 – Double Validation croisée avec $n = 4$.

CHAPITRE VII

Conclusion

NOUS nous sommes intéressés durant ce travail de thèse aux interfaces cerveau machine non invasives utilisant des mesures de l'activité cérébrale par électroencéphalographie (EEG) et plus précisément aux ICM basées sur l'imagination d'actions motrices. Quand nous avons commencé ce travail, l'équipe Athéna débutait son activité de recherche dans le domaine des ICM. Une de nos contributions majeures est d'avoir mis en place un système permettant la réalisation et l'étude des ICM. À ce jour, une part grandissante de l'équipe Athéna travaille sur les ICM et cette étude est devenue un thème de recherche à part entière pour l'équipe.

Une des principales limites des ICM utilisant l'EEG est, à l'heure actuelle, la faiblesse du taux de transfert d'information. Cette limite provient en grande partie de la faible quantité d'information se trouvant dans le signal mesuré et de la faiblesse du rapport signal sur bruit. Cette pauvreté du signal est due au fait que les capteurs EEG sont relativement éloignés des sources d'intérêt et que le crâne limite fortement la précision des mesures.

Nous avons, durant le début de ce travail de thèse, utilisé deux types de méthodes pour reconstruire en temps réel l'activité des sources corticales à l'origine du signal mesuré. Ce travail fait l'objet du chapitre V de ce manuscrit. De telles méthodes pourraient rendre l'information présente dans le signal mesuré plus accessible et ainsi améliorer le fonctionnement du transducteur. Cependant, la reconstruction de l'activité cérébrale est une tâche complexe et une solution fidèle

est particulièrement difficile à obtenir en essai par essai à cause de la faiblesse du rapport signal sur bruit. Il n'est donc pas certain qu'elle puisse être réalisée avec suffisamment de précision pour réellement augmenter les capacités des ICM. Le travail effectué montre une nouvelle fois qu'il est possible d'accomplir une reconstruction de sources en temps réel, mais ne permet pas de conclure sur l'utilité réelle de telles méthodes pour les ICM. De notre point de vue, seule la réalisation d'une "BCI competition" permettrait de mesurer objectivement cet apport.

Nous avons par la suite focalisé notre attention sur l'étude de l'imagination des mouvements brefs (qui font l'objet du chapitre III de ce manuscrit). De tels mouvements produisent généralement une désynchronisation des rythmes μ et β durant la réalisation ou l'imagination (appelés ERD pour *event related desynchronisation*) et sont suivis d'une augmentation de puissance supérieure à la normale des mêmes rythmes (appelés ERS pour *event related synchronisation* et parfois rebond β). Peu d'ICM utilisent à l'heure actuelle l'information contenue dans l'ERS qui suit l'imagination des mouvements. L'étude détaillée des mouvements brefs montre que les rythmes μ et β ont une cartographie complexe dans le plan temps-fréquence et qui varie fortement selon les sujets. Nous avons montré qu'il est possible de détecter et de différencier l'imagination de plusieurs actions motrices brèves grâce à l'ERS et l'ERD. Une première ICM asynchrone permettant la navigation spatiale dans un environnement 3D a été réalisée grâce à l'utilisation du rebond β . Une nouvelle ICM permettant l'utilisation de plusieurs commandes est en cours de réalisation. Cette ICM permettra de commander plusieurs boutons en imaginant la réalisation de plusieurs tâches motrices.

Il est possible d'imaginer un grand nombre de tâches motrices (ou cognitives) différentes pour contrôler une ICM. Certaines tâches étant mieux discriminées que d'autres, il est important d'utiliser des tâches adéquates. Malheureusement les tâches qui permettent le meilleur contrôle ne sont pas identiques pour tous les sujets. Nous avons donc développé un algorithme, utilisant la théorie des bandits stochastiques, qui sélectionne automatiquement et efficacement une tâche adaptée au sujet. Le chapitre IV de ce manuscrit est consacré à cet algorithme. En éliminant rapidement les tâches sub-optimales, cet algorithme permet d'explorer un large ensemble de tâches tout en garantissant une estimation fiable des tâches les plus performantes.

Nous avons développé une boîte à outils Matlab, appelée *MoffBCI*, pour l'analyse hors-ligne des ICM. Le fonctionnement de cette boîte à outils est détaillé dans le chapitre VI de ce manuscrit. Nous l'avons spécialement conçue pour la comparaison de méthodes et l'ajustement automatique de paramètres. Nous avons également contribué au logiciel opensource OpenViBE qui permet de réaliser des ICM. OpenViBE étant dédié au fonctionnement en-ligne des ICM, *MoffBCI* en est un complément intéressant pour améliorer et optimiser les ICM.

PERSPECTIVES

Le travail que nous avons effectué ouvre plusieurs perspectives sur différents aspects des ICM.

Utilisation de l’imagination motrice brève. Bien que la période suivant l’imagination motrice (ERS) ait été utilisée pour réaliser un interrupteur commandé par la pensée, nous sommes les premiers à avoir étudié en détail l’imagination des mouvements brefs. Ce travail a permis de montrer que l’ERS peut être utilisée pour distinguer l’imagination de plusieurs mouvements brefs, ce qui est sûrement la contribution principale de cette thèse, et ouvre donc de nombreuses perspectives :

- Amélioration de la fiabilité de l’interface de navigation spatiale. Cette ICM, décrite dans la section 2 du chapitre III, utilise actuellement un bouton commandé par le rebond β . Nous avons montré qu’une utilisation conjointe de l’ERD et de l’ERS dans le μ et le β permet de meilleures performances.
- Réalisation de boutons multiples commandés par la pensée. La distinction de plusieurs tâches d’imagination motrices brèves permet d’augmenter le nombre de commandes qu’utilise l’interface de navigation spatiale mais aussi de créer de nouvelles ICM.
- Le jeu “RockInvaders”, que nous avons développé (voir section 3.6.2 du chapitre III), permet de réaliser des expériences EEG stimulantes pour le sujet tout en permettant d’exploiter les données enregistrées pour effectuer un apprentissage. Une prochaine série d’expériences en-ligne, réalisées par Eoin Thomas, devrait inclure une session d’utilisation de “RockInvaders”.
- Si seule la période d’imagination est utilisée pour la classification, la période suivant l’imagination du mouvement pourrait être utilisée pour la labélisation des mouvements. Ceci permet de continuer à effectuer un apprentissage durant l’utilisation de l’ICM (voir section 3.6.2 du chapitre III pour plus de détails).

Sélection de tâche. L’algorithme que nous avons conçu pour la sélection de tâches permet de sélectionner la tâche qui est la plus facilement détectable au milieu d’une période de repos de l’utilisateur. Cet algorithme peut être amélioré de deux façons :

- Il peut être généralisé à la sélection d’une paire de tâche, voire à un ensemble optimal de tâches. Le principe de cette généralisation est décrit à la fin du chapitre IV.
- Une amélioration plus complexe consiste à optimiser les features utilisées au cours de la sélection de tâche. Le nombre d’essais correspondant à chaque tâche étant très réduit au commencement de l’expérience, il est probable qu’il faille commencer par optimiser les features pour l’ensemble des tâches puis utiliser des features dédiées aux tâches les plus performantes dans un

second temps.

Reconstruction de sources pour les ICM. Comme expliqué à la fin du chapitre V, de notre point de vue, seule l'organisation d'une "BCI competition" permettra de juger de l'utilité réelle de la reconstruction de l'activité cérébrale pour les ICM. L'équipe Athéna disposait déjà d'une forte expertise du problème direct, le travail que nous avons réalisé a permis d'étudier la résolution du problème inverse en temps réel pour les ICM. Pour ces raisons, l'équipe Athéna devrait prochainement commencer une campagne d'acquisitions d'expériences EEG permettant la réalisation d'une "BCI competition" sur la reconstruction de sources.

Annexes

Exemple d'utilisation de *MoffBCI*

1 GÉNÉRATION D'UN JEU DE DONNÉES ARTIFICIEL

À titre d'exemple nous allons générer un petit jeu de données simulant de l'imagination motrice. Les données sont composées de *nb_trials* = 100 essais durant lesquels un utilisateur effectuerait deux types de mouvements différents :

- le 1^{er} mouvement génère des oscillations à 2 et 5Hz sur les électrodes 1 et 2.
- le 2^e mouvement génère des oscillations à 13 et 17Hz sur les électrodes 3 et 4.

```

1 % Parameters:
2 nb_trials = 200; % Number of trials
3 nb_electrodes = 16; % Number of electrodes
4 trial_length = 0.5; % Length of the trials in seconds
5 Fech = 512; % sampling frequency
6 SNR = 1/20; % Signal over noise ratio
7
8 nb_samples = trial_length*Fech;
9 T = (1:nb_samples)/Fech;
10
11 global Data Y Data_info
12 Data_info.f_ech = Fech;
13 Y = randi([1 2],nb_trials,1);
14 Data = cell(nb_trials,1);
15 for i=1:nb_trials
16     Data{i} = rand(nb_electrodes,nb_samples);
17     if Y(i) == 1
18         Data{i}(1,:) = Data{i}(1,:) + sin(T*2*pi * 2)*SNR;
19         Data{i}(2,:) = Data{i}(2,:) + sin(T*2*pi * 5)*SNR;
20     else
21         Data{i}(3,:) = Data{i}(3,:) + sin(T*2*pi * 13)*SNR;
22         Data{i}(4,:) = Data{i}(4,:) + sin(T*2*pi * 17)*SNR;
23     end
24 end

```

2 ÉCRITURE D'UNE CHAÎNE DE TRAITEMENT

On définit la liste des fonctions constituant la chaîne de traitement et les paramètres à utiliser. On va essayer différentes listes de fréquences et vérifier que les méthodes de sélection automatique des paramètres permettent bien de retrouver la bonne.

```

1 % Function list :
2 %%%%%%%%%%
3 ProcessStruct.functions(1).name = 'features_extraction';
4 ProcessStruct.functions(1).global_in = {'Data'; 'Data_info'};
5 ProcessStruct.functions(1).global_out = {'Features'};
6
7 ProcessStruct.functions(2).name = 'split_train_test';
8 ProcessStruct.functions(2).global_in = {'Features'; 'Y'; 'Data_info'};
9 ProcessStruct.functions(2).global_out = {'X_train'; 'X_test'; 'Y_train'; 'Y_test'};
10
11 ProcessStruct.functions(3).name = 'features_selection';
12 ProcessStruct.functions(3).global_in = {'X_train'; 'X_test'; 'Y_train'};
13 ProcessStruct.functions(3).global_out = {'X_train'; 'X_test'};
14
15 ProcessStruct.functions(4).name = 'classification';
16 ProcessStruct.functions(4).global_in = {'X_train'; 'X_test'; 'Y_train'; 'Y_test'};
17 ProcessStruct.functions(4).global_out = {};
18
19 % Parameters :
20 %%%%%%%%%%
21
22 %% for the features extraction:
23 ProcessStruct.feats_extract_method = 'Joan'; % name of the methode
24
25 ProcessStruct.feats_extract_Elec = [1:16]'; % electrodes from which features are extracted
26 % frequencies at which features are extracted. We will test two sets of frequencies
27 ProcessStruct.feats_extract_Freq.function = 'features_extraction';
28 ProcessStruct.feats_extract_Freq.values = {[2 5 13 17]'; [3 6 14 18]'};
29
30 %% for the features selection: (total number = nb_electrodes x nb_frequencies)
31 ProcessStruct.feats_select_nb_feats.values = {1 4 8 10 12}; % number of features to use
32 ProcessStruct.feats_select_nb_feats.function = 'features_selection';
33 ProcessStruct.feats_select_alpha = 0.3; % parameter for the recursive features selection
34
35 %% for the classification
36 ProcessStruct.classif_method = 'SVM';
37 ProcessStruct.classif_kernel = 'linear';
38 ProcessStruct.classif_lambda = 1;

```

3 ÉVALUATION DE LA CHAÎNE DE TRAITEMENT

3.1 Évaluation standard

On va effectuer l'apprentissage sur les 120 premiers essais puis effectuer le test sur les 80 essais restants.

```

1 ProcessStruct.split_train_test.ind_train = 1:120;
2 ProcessStruct.split_train_test.ind_test = 121:200;
3
4 global ToyResults
5 ProcessStruct.TreeName = 'ToyResults'; % The name of the tree where the results are saved.
6 run_functions(ProcessStruct); % Evaluation of the processing chaîne

```

3.2 Validation croisée

On définit une partitions des données en 5 ensembles qui serviront alternativement pour l'apprentissage ou le test des performances. Il suffit ensuite d'utiliser la fonction *simple_cross_validation*.

```
1 Data_info.ind_train_test = {1:40 41:80 81:120 121:160 161:200};
2 [best_parameters , TreeResults , BestResult] = simple_cross_validation(ProcessStruct);
3 plot_tree(TreeResults);
```

Les résultats obtenus sont représenté sur la figure A.1.

3.3 Évaluation tripartite ou 3-way data split

On va réutiliser les sous ensembles définis précédemment mais s'en servir pour effectuer une partition des données en trois ensembles :

- L'ensemble d'apprentissage (ici constitué des sous ensembles 1 et 2) : permet d'effectuer un 1^{er} apprentissage avec différentes valeurs pour les paramètres.
- L'ensemble d'évaluation (le sous ensemble 3) : permet d'évaluer les paramètres.
- L'ensemble de test (les sous ensembles 4 et 5) : permet d'évaluer les performances finales après avoir fait un nouvel apprentissage sur les ensembles d'apprentissage et d'évaluation.

```
1 Data_info.ind_train_test = {1:40 41:80 81:120 121:160 161:200};
2 [best_parameters , EvalTree , ResultTree] = three_way_split(ProcessStruct , {1:2 , 3 , 4:5});
3 plot_tree(EvalTree);
4 figure(2);
5 plot_tree(ResultTree);
```

La figure A.2 représente l'arbre des résultats obtenus sur l'ensemble d'évaluation puis les résultats finaux avec les paramètres qui ont été sélectionnés.

3.4 Double validation croisée

Pour la double validation croisée les résultats sont sauvegardés dans un sous dossier en deux variables :

- **LTree_NomDeLArbre**, comporte une liste d'arbres donnant les résultats des évaluations des paramètres.
- **ResTree_NomDeLArbre**, comporte les résultats finaux sous forme d'un arbre unique. Idéalement, si tous les seconds niveaux de validation croisée ont abouti à sélectionner les mêmes paramètres, cet arbre n'a qu'une seule feuille (voir figure A.3).

```
1 Data_info.ind_train_test = {1:40 41:80 81:120 121:160 161:200};
2 ProcessStruct.TreeName = 'ToyResults';
3
4 cross_validation(ProcessStruct);
5
```

```
6 load TreeSave/LTree_ToyResults
7 plot_tree(LTree_ToyResults{1})
8
9 load TreeSave/ResTree_ToyResults
10 figure(2);
11 plot_tree(ResTree_ToyResults)
```

4 RÉSULTATS

Comme on peut le constater sur les différentes figures ci-dessous, les meilleures performances sont bien obtenues quand on utilise les fréquences ayant servi à générer les données. D'autre part l'utilisation de la sélection de features permet d'augmenter légèrement les performances.

Ces deux paramètres peuvent être réglés de façon automatique si on utilise l'évaluation tripartite ou la double validation croisée.

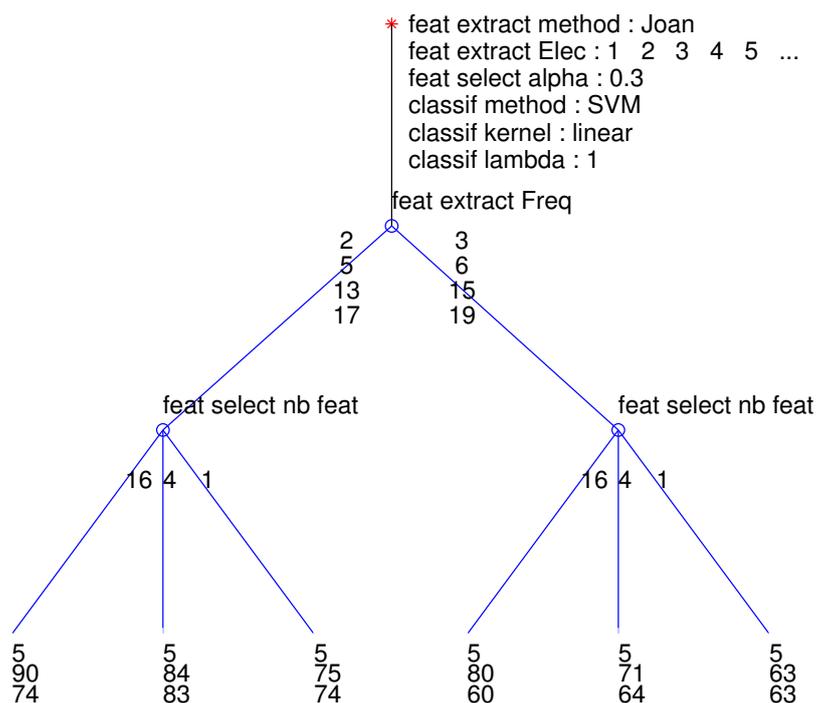


Fig. A.1 – Arbre des résultats de la chaîne de traitement évaluée par la validation croisée. Les paramètres affichés en haut sont fixés. Les autres constituent les nœuds de l'arbre. Trois nombres sont représentés sous chaque feuille : le premier est le nombre de résultats qui sont regroupés dans cette feuille (pour la validation croisée il doit être identique pour chaque feuille et correspond au nombre d'ensembles). Le deuxième donne les performances en pourcentage de bien classés sur les données d'apprentissage et le troisième sur les données de teste.

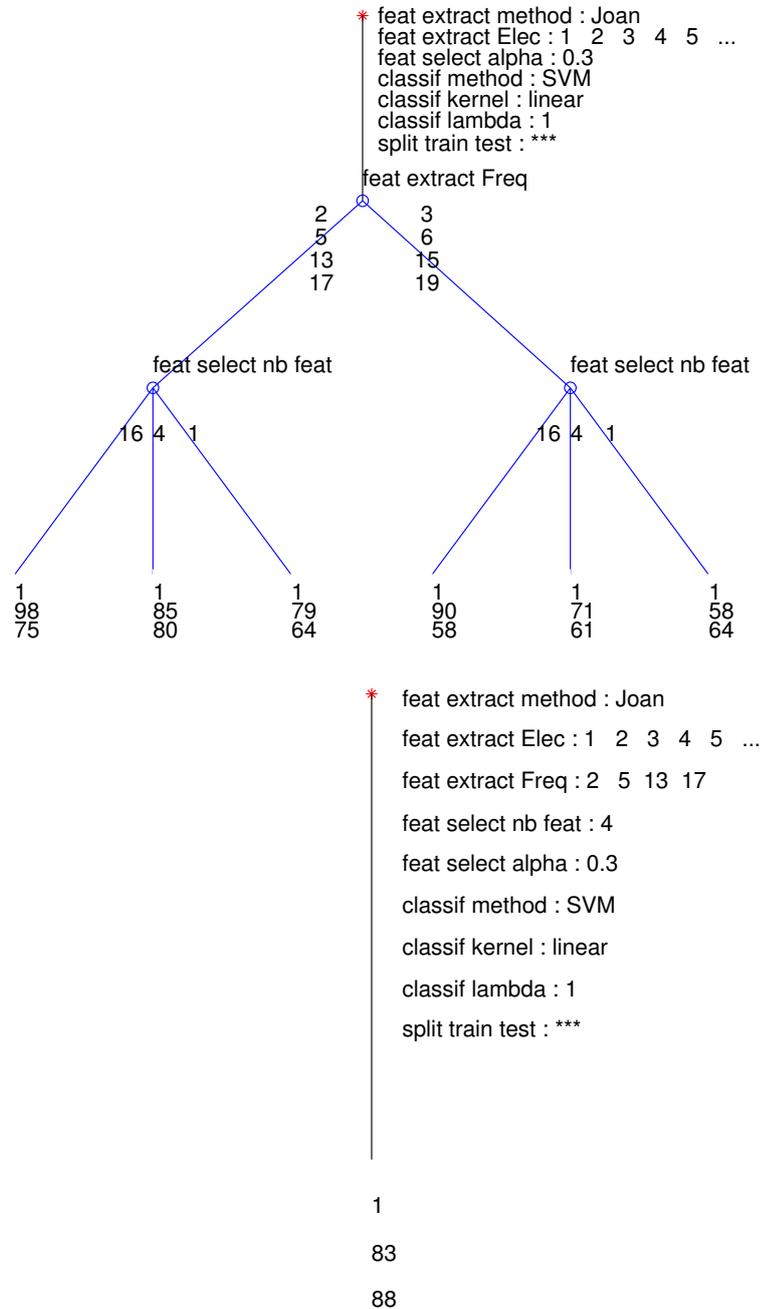
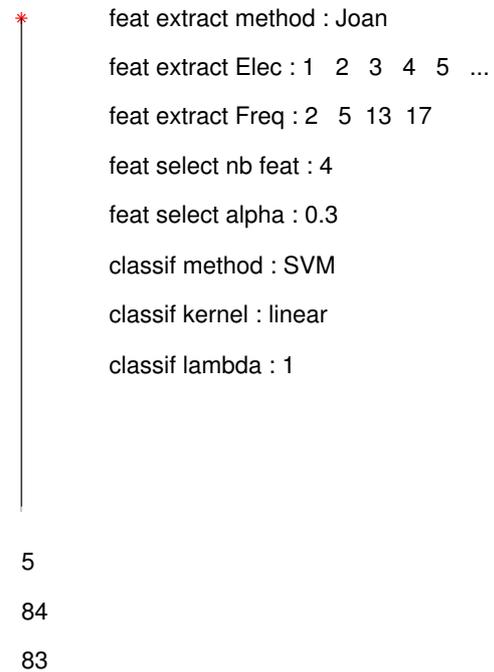


Fig. A.2 – Arbres obtenus lors de l'évaluation tripartite de la chaîne de traitement. L'arbre supérieur représente l'évaluation des différents paramètres. La feuille ayant la meilleure performance sur les données d'apprentissage (80%) correspond à l'utilisation des fréquences 2, 5, 13, et 17Hz et en sélectionnant 4 features. C'est ces paramètres qui sont utilisés pour le nouvel apprentissage et le test sur le troisième ensemble de données. Les résultats finaux sont représentés par un arbre ne disposant que d'une feuille puisqu'aucun paramètre ne varie. On remarquera que la performance finale sur l'ensemble d'apprentissage (83%) est inférieure à celle des données test (88%), cela montre l'imprécision obtenue quand l'ensemble de test est trop réduit.



```

*
feat extract method : Joan
feat extract Elec : 1 2 3 4 5 ...
feat extract Freq : 2 5 13 17
feat select nb feat : 4
feat select alpha : 0.3
classif method : SVM
classif kernel : linear
classif lambda : 1

5
84
83

```

Fig. A.3 – Résultats finaux obtenus par la double validation croisée. Sur chaque ensemble d'apprentissage, les mêmes paramètres sont sélectionnés par l'étape de validation croisée interne. L'arbre de résultats final ne comporte donc qu'une feuille où se trouvent 5 résultats moyennés.

Écriture d'une fonction de traitement de *MoffBCI*

L'écriture d'une fonction pour *MoffBCI* est extrêmement simple. La seule contrainte est que le premier argument de la fonction soit une structure dans laquelle sont rangés les paramètres.

1 EXEMPLE : LA FONCTION “TEMPORAL FILTER”

1.1 Définition

Comme son nom l'indique le but de la fonction “temporal filter” est d'appliquer un filtre temporel sur les données.

1.2 Les entrées et sorties

En plus du *StructParameter* la fonction prend en entrées les données *Data* ainsi qu'une structure *Data_info* dans laquelle est stockée la fréquence d'échantillonnage des données :

$$F_{ech} = Data_info.f_ech$$

Les données peuvent être une simple matrice de taille $[N_{capteur} \times N_{echantillons}]$ ou un tableau (*cell*) de telles matrices dans le cas où les données sont découpées en essais.

$$Data = [N_{capteur} \times N_{echantillons}] \text{ ou } Data = \{[N_{capteur} \times N_{echantillons}]\}_{N_{essais}}$$

La sortie est constituée des données filtrées que nous appellerons aussi *Data*.

1.3 Les paramètres

Les paramètres de la fonction sont :

- le type de filtre : *filter_type* = "band_pass"
- la, ou les, fréquences de coupure : *frequencies* = [4, 40]
- l'ordre du filtre : *order* = 4

Le code de la fonction temporal_filter

```

1  function Data = temporal_filter(para , Data , Data_info)
2  % Filters the input data with a butterworth filter
3  %
4  % inputs :
5  % Data : a (Nsensor x Nsamples) matrix or a cell of such matrix
6  % Data_info.f_ech the sampling rate frequency
7  %
8  % output :
9  % Data : a Ntrials cell containing (Nsensor x Nsamples) matrix
10
11 % Parameters :
12 % filter_type = "band_pass, band_stop, low_cut, high_cut"
13 % frequencies = (4,40)
14 % order = 4
15
16 f_ech = Data_info.f_ech;
17
18 % the parameters that must be load from para :
19 para_names = {...
20     'temporal_filter_filter',      'filter_type', 'band_pass'; ...
21     'temporal_filter_frequencies', 'frequencies', [2 45]; ...
22     'temporal_filter_order',      'order',      4 ...
23 };
24 load_parameters_from_para %the script that loads the parameters
25
26 switch lower(filter_type)
27     case 'band_pass'
28         [z, p, k] = butter(order, frequencies(1:2)/f_ech*2);
29     case 'band_stop'
30         [z, p, k] = butter(order, frequencies(1:2)/f_ech*2, 'stop');
31     case 'low_cut'
32         [z, p, k] = butter(order, frequencies(1)/f_ech*2, 'high');
33     case 'high_cut'
34         [z, p, k] = butter(order, frequencies(1)/f_ech*2, 'low');
35     otherwise
36         error(['temporal_filter: unknown filter.\n'...
37             'filters are: band_pass band_stop low_cut high_cut'])
38 end
39
40 [sos ,g]=zp2sos(z,p,k);
41 Hd=dfilt.df2sos(sos ,g);
42
43 if iscell(Data)
44     for tr = 1:length(Data);
45         Data{tr} = filter(Hd,Data{tr},2);
46     end
47 else
48     Data = filter(Hd,Data,2);
49 end

```

Bibliographie

- [**Ahn et al., 2010**] AHN, M., HONG, J. H. et JUN, S. C. (2010). Source space based brain computer interface. In SUPEK, S., SUŠAC, A. et MAGJAREVIC, R., éditeurs : *17th International Conference on Biomagnetism Advances in Biomagnetism – Biomag2010*, volume 28 de *IFMBE Proceedings*, pages 366–369. Springer Berlin Heidelberg. 149, 157
- [**Ahn et al., 2012**] AHN, M., HONG, J. H. et JUN, S. C. (2012). Feasibility of approaches combining sensor and source features in brain–computer interface. *Journal of Neuroscience Methods*, 204(1):168 – 178. 64, 65, 143, 149, 157
- [**Ang et al., 2009**] ANG, K. K., CHIN, Z. Y., ZHANG, H. et GUAN, C. (2009). Robust filter bank common spatial pattern (rfbcsp) in motor-imagery-based brain-computer interface. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 578–581. 60, 61
- [**Auer et al., 2002**] AUER, P., CESA-BIANCHI, N. et FISCHER, P. (2002). Finite time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3): 235–256. 118, 120
- [**Babiloni et al., 2007**] BABILONI, F., CINCOTTI, F., MARCIANI, M., SALINARI, S., ASTOLFI, L., TOCCI, A., ALOISE, F., DE VICO FALLANI, F., BUFALARI, S. et MATTIA, D. (2007). The estimation of cortical activity for brain-computer interface: applications in a domotic context. *Intell. Neuroscience*, 2007:11–11. 63, 64, 65, 145, 157
- [**Bai et al., 2008**] BAI, O., LIN, P., VORBACH, S., FLOETER, M. K., HATTORI, N. et HALLETT, M. (2008). A high performance sensorimotor beta rhythm-based brain–computer interface associated with human natural motor behavior. *Journal of Neural Engineering*, 5(1):24. 83, 92
- [**Baillet et al., 2001**] BAILLET, S., MOSHER, J. et LEAHY, R. (2001). Electromagnetic brain mapping. *IEEE Signal Processing Magazine*, 18(6):14–30. 62, 140, 141
- [**Barachant et al., 2011**] BARACHANT, A., BONNET, S., CONGEDO, M. et JUTTEN, C. (2011). A brain-switch using riemannian geometry. In *Proceedings of the 5th International BCI Conference 2011*, pages 64–67, Graz, Autriche. 54, 61
- [**Barinaga, 1995**] BARINAGA, M. (1995). Remapping the motor cortex. *Science*, 268(5218):1696–1698. 42
- [**Bayliss et Ballard, 2000**] BAYLISS, J. et BALLARD, D. (2000). A virtual reality testbed for brain-computer interface research. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):188 –190. 16

- [**Belouchrani et al., 1997**] BELOUCHRANI, A., ABED-MERAÏM, K., CARDOSO, J.-F. et MOULINES, E. (1997). A blind source separation technique using second-order statistics. *Signal Processing, IEEE Transactions on*, 45(2):434–444. 61
- [**Besserve, 2007**] BESSERVE, M. (2007). *Analyse de la dynamique neuronale pour les Interfaces Cerveau-Machines : un retour aux sources*. Thèse de doctorat, ED STITS. 13, 145, 157
- [**Besserve et al., 2008**] BESSERVE, M., MARTINERIE, J. et GARNERO, L. (2008). Non-invasive classification of cortical activities for brain computer interface: A variable selection approach. In *Biomedical Imaging: From Nano to Macro, ISBI*, pages 1063–1066. 152
- [**Besserve et al., 2011**] BESSERVE, M., MARTINERIE, J. et GARNERO, L. (2011). Improving quantification of functional networks with EEG inverse problem: Evidence from a decoding point of view. *NeuroImage*, 55(4):1536 – 1547. 45, 64, 65, 117, 145, 157
- [**Birbaumer et al., 2000**] BIRBAUMER, N., KUBLER, A., GHANAYIM, N., HINTERBERGER, T., PERELMOUTER, J., KAISER, J., IVERSEN, I., KOTCHOUBEY, B., NEUMANN, N. et FLOR, H. (2000). The thought translation device (ttD) for completely paralyzed patients. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):190–193. 39
- [**Blankertz et al., 2006**] BLANKERTZ, B., MULLER, K.-R., KRUSIENSKI, D., SCHALK, G., WOLPAW, J., SCHLOGL, A., PFURTSCHELLER, G., MILLAN, J., SCHRODER, M. et BIRBAUMER, N. (2006). The bci competition iii: validating alternative approaches to actual bci problems. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):153–159. 59, 61
- [**Blum et Langley, 1997**] BLUM, A. L. et LANGLEY, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1–2):245–271. 152
- [**Bozorgzadeh et al., 2000**] BOZORGZADEH, Z., BIRCH, G. et MASON, S. (2000). The LF-ASD brain computer interface: on-line identification of imagined finger flexions in the spontaneous EEG of able-bodied subjects. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, volume 6, pages 2385–2388 vol.4. 55
- [**Brunner et al., 2011**] BRUNNER, C., ANDREONI, G., BIANCHI, L. and Blankertz, B., BREITWIESER, C., KANOH, S., A. KOTHE, C., LÉCUYER, A., MAKEIG, S., MELLINGER, J., PEREGO, P., RENARD, Y., SCHALK, G., SUSILA, I. P., VENTHUR, B. et MÜLLER-PUTZ, G. R. (2011). BCI software platforms. 163
- [**Brunner et al., 2010**] BRUNNER, P., JOSHI, S., BRISKIN, S., WOLPAW, J. R., BISCHOF, H. et SCHALK, G. (2010). Does the 'p300' speller depend on eye gaze? *Journal of Neural Engineering*, 7(5):056013. 34

- [**Capon, 1969**] CAPON, J. (1969). High-resolution frequency-wavenumber spectrum analysis. *Proceedings of the IEEE*, 57(8):1408 – 1418. 146
- [**Congedo et al., 2006**] CONGEDO, M., LOTTE, F. et LÉCUYER, A. (2006). Classification of movement intention by spatially filtered electromagnetic inverse solutions. *Physics in Medicine and Biology*, 51:1971–1989. 63, 157
- [**Coyle et al., 2007**] COYLE, S. M., WARD, T. E. et MARKHAM, C. M. (2007). Brain–computer interface using a simplified functional near-infrared spectroscopy system. *Journal of Neural Engineering*, 4(3):219. 54
- [**Dal Seno et al., 2010**] DAL SENO, B., MATTEUCCI, M. et MAINARDI, L. (2010). Online detection of p300 and error potentials in a bci speller. *Intell. Neuroscience*, 2010:11:1–11:1. 37
- [**Dalal et al., 2006**] DALAL, S., SEKIHARA, K. et NAGARAJAN, S. (2006). Modified beamformers for coherent source region suppression. *Biomedical Engineering, IEEE Transactions on*, 53(7):1357 –1363. 150
- [**Delorme et al., 2011**] DELORME, A., MULLEN, T., KOTHE, C., ACAR, Z. A., BIGDELY-SHAMLO, N., VANKOV, A. et MAKEIG, S. (2011). Eeglab, sift, nft, beilab, and erica: new tools for advanced eeg processing. *Intell. Neuroscience*, 2011:10:10–10:10. 163
- [**Dickhaus et al., 2009**] DICKHAUS, T., SANNELLI, C., MÜLLER, K.-R., CURIO, G. et BLANKERTZ, B. (2009). Predicting bci performance to study bci illiteracy. *BMC Neuroscience*, 10:1–2. 109, 117
- [**Dobrea et al., 2010**] DOBREA, M., DOBREA, D. et ALEXA, D. (2010). Spectral EEG features and tasks selection process: Some considerations toward BCI applications. In *Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on*, pages 150 –155. 56, 117, 127, 135
- [**Dobrea et Dobrea, 2009**] DOBREA, M.-C. et DOBREA, D. (2009). The selection of proper discriminative cognitive tasks - a necessary prerequisite in high-quality BCI applications. In *Applied Sciences in Biomedical and Communication Technologies, 2009. ISABEL 2009. 2nd International Symposium on*, pages 1 –6. 56, 57, 117, 127, 135
- [**Doležal et al., 2011**] DOLEŽAL, J., ŠTĀSTNY, J. et SOVKA, P. (2011). Exploiting temporal context in high-resolution movement-related EEG classification. *Radioengineering*, 20(3):666–676. 110
- [**Donchin et al., 2000**] DONCHIN, E., SPENCER, K. et WIJESINGHE, R. (2000). The mental prosthesis: assessing the speed of a p300-based brain-computer interface. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):174 –179. 13, 16
- [**Dornhege et al., 2006**] DORNHEGE, G., BLANKERTZ, B., KRAULEDAT, M., LOSCH, F., CURIO, G. et MÜLLER, K.-R. (2006). Optimizing spatio-temporal

- filters for improving BCI. *In in Advances in Neural Inf. Proc. Systems (NIPS 05)*, volume 18, pages 315–322. 60
- [**Duncan et al., 2009**] DUNCAN, C. C., BARRY, R. J., CONNOLLY, J. F., FISCHER, C., MICHIE, P. T., NÄÄTÄNEN, R., POLICH, J., REINVANG, I. et PETTEN, C. V. (2009). Event-related potentials in clinical research: Guidelines for eliciting, recording, and quantifying mismatch negativity, P300, and N400. *Clinical Neurophysiology*, 120(11):1883 – 1908. 30
- [**Elbert et al., 1980**] ELBERT, T., ROCKSTROH, B., LUTZENBERGER, W. et BIRBAUMER, N. (1980). Biofeedback of slow cortical potentials. i. *Electroencephalography and Clinical Neurophysiology*, 48(3):293 – 301. 39
- [**Farwell et Donchin, 1988**] FARWELL, L. et DONCHIN, E. (1988). Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6):510 – 523. 30, 32, 49
- [**Fatourechi et al., 2008**] FATOURECHI, M., WARD, R., MASON, S., HUGGINS, J., SCHLOGL, A. et BIRCH, G. (2008). Comparison of evaluation metrics in classification applications with imbalanced datasets. *In Machine Learning and Applications, 2008. ICMLA '08. Seventh International Conference on*, pages 777–782. IEEE. 55
- [**Feldman et Griffiths, 1991**] FELDMAN, D. et GRIFFITHS, L. (1991). A constraint projection approach for robust adaptive beamforming. *In Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*, pages 1381 –1384 vol.2. 151
- [**Ferrez et Millán, 2008**] FERREZ, P. et MILLÁN, J. (2008). Simultaneous real-time detection of motor imagery and error-related potentials for improved BCI accuracy. *In Proceedings of the 4th International Brain-Computer Interface Workshop and Training Course*, pages 197–202. Citeseer. 37
- [**Ferrez et Millan, 2008**] FERREZ, P. et MILLAN, J. d. R. (2008). Error-related EEG potentials generated during simulated brain-computer interaction. *IEEE Trans. Biomed. Engin.*, 55(3). 37
- [**Franceschini et al., 2000**] FRANCESCHINI, M. A., TORONOV, V., FILIACI, M., GRATTON, E. et FANTINI, S. (2000). On-line optical imaging of the human brain with 160-ms temporal resolution. *Optics Express*, 6:49. 28
- [**Fruitet et Clerc, 2009**] FRUITET, J. et CLERC, M. (2009). Comparaison de problèmes inverses pour la classification d’activité cérébrale en temps réel. *In proceedings of GRETSI*. 44, 145, 149
- [**Fruitet et Clerc, 2010**] FRUITET, J. et CLERC, M. (2010). Reconstruction of cortical sources activities for online classification of electroencephalographic signals. *In EMBC: Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. 145, 149

- [**Fruitet et al., 2011**] FRUITET, J., CLERC, M. et PAPADOPOULO, T. (2011). Preliminary study for an hybrid BCI using sensorimotor rhythms and Beta rebound. In *International Journal of Bioelectromagnetism*, volume 13, pages 70–71. 56, 83, 91, 111
- [**Fruitet et al., 2010**] FRUITET, J., MCFARLAND, D. J. et WOLPAW, J. R. (2010). A comparison of regression techniques for a two-dimensional sensorimotor rhythm-based brain-computer interface. *Journal of Neural Engineering*, 7(1). 14, 77, 82, 153
- [**Fukunuga, 1990**] FUKUNUGA, K. (1990). *Introduction to statistical pattern recognition*. Academic Press. 59
- [**Galán et al., 2008**] GALÁN, F., NUTTIN, M., LEW, E., FERREZ, P., VANACKER, G., PHILIPS, J. et MILLÁN, J. (2008). A brain-actuated wheelchair: Asynchronous and non-invasive brain-computer interfaces for continuous control of robots. *Clinical Neurophysiology*, 119(9):2159–2169. 49, 53
- [**Grafton et al., 1996**] GRAFTON, S. T., ARBIB, M. A., FADIGA, L. et RIZZOLATTI, G. (1996). Localization of grasp representations in humans by positron emission tomography. *Experimental Brain Research*, 112:103–111. 10.1007/BF00227183. 40
- [**Graimann et al., 2002**] GRAIMANN, B., HUGGINS, J., LEVINE, S. et PFURTSCHELLER, G. (2002). Visualization of significant ERD/ERS patterns in multi-channel EEG and ECoG data. *Clinical Neurophysiology*, 113(1):43 – 47. 97
- [**Gramfort et al., 2011**] GRAMFORT, A., PAPADOPOULO, T., OLIVI, E. et CLERC, M. (2011). Forward field computation with OpenMEEG. *Computational Intelligence and Neuroscience*. 143
- [**Grave de Peralta Menendez et al., 2005**] GRAVE DE PERALTA MENENDEZ, R., GONZÁLEZ ANDINO, S., PEREZ, L., FERREZ, P. W. et MILLÁN, J. d. R. (2005). Non-invasive estimation of local field potentials for neuroprosthesis control. *Cognitive Processing*, 6:59–64. 10.1007/s10339-004-0043-x. 63, 64, 65, 143, 157
- [**Grech et al., 2008**] GRECH, R., CASSAR, T., MUSCAT, J., CAMILLERI, K., FABRI, S., ZERVAKIS, M., XANTHOPOULOS, P., SAKKALIS, V. et VANRUMSTE, B. (2008). Review on solving the inverse problem in eeg source analysis. *Journal of NeuroEngineering and Rehabilitation*, 5(1):25. 26
- [**Gross et Ioannides, 1999**] GROSS, J. et IOANNIDES, A. (1999). Linear transformations of data space in MEG. *Physics in Medicine and Biology*, 44:2081–2097. 148, 150
- [**Grosse-Wentrup et Buss, 2008**] GROSSE-WENTRUP, M. et BUSS, M. (2008). Multiclass common spatial patterns and information theoretic feature extraction. *Biomedical Engineering, IEEE Transactions on*, 55(8):1991 –2000. 60

- [**Grosse-Wentrup et al., 2009**] GROSSE-WENTRUP, M., LIEFHOLD, C., GRAMANN, K. et BUSS, M. (2009). Beamforming in noninvasive brain-computer interfaces. *Biomedical Engineering, IEEE Transactions on*, 56(4):1209–1219. 64, 65, 143, 149, 157
- [**Hämäläinen et al., 1993**] HÄMÄLÄINEN, M., HARI, R., IIMONIEMI, R. J., KNUUTILA, J. et LOUNASMAA, O. V. (1993). Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain. *Reviews of Modern Physics*, 65(2):413–497. 27
- [**Hill et al., 2006**] HILL, N., LAL, T., SCHRÖDER, M., HINTERBERGER, T., WIDMAN, G., ELGER, C., SCHÖLKOPF, B. et BIRBAUMER, N. (2006). Classifying event-related desynchronization in EEG, ECoG and MEG signals. In *Pattern Recognition*, volume 4174 de *Lecture Notes in Computer Science*, pages 404–413. Springer Berlin / Heidelberg. 139
- [**Hochberg et al., 2006**] HOCHBERG, L. R., SERRUYA, M. D., FRIEHS, G. M., MUKAND, J. A., SALEH, M., CAPLAN, A. H., BRANNER, A., CHEN, D., PENN, R. D. et DONOGHUE, J. P. a. (2006). Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442:164–171. 24
- [**Hyvarinen et al., 2001**] HYVARINEN, A., J. KARHUNEN, J. et P., O. (2001). *Independent Component Analysis*. Ser. New York: Wiley Interscience. 61
- [**Hyvärinen et Oja, 2000**] HYVÄRINEN, A. et OJA, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, 13(4–5):411–430. 61
- [**Jasper et Andrew, 1938**] JASPER, H. et ANDREW, H. (1938). Electroencephalography iii. normal differentiation of occipital and precentral regions in man. *Arch Neurol Psychiatry*. 97
- [**Jeon et al., 2011**] JEON, Y., NAM, C., KIM, Y. et WHANG, M. (2011). Event-related (de) synchronization (ERD/ERS) during motor imagery tasks: Implications for brain-computer interfaces. *International Journal of Industrial Ergonomics*. 92
- [**Joyce et al., 2004**] JOYCE, C. A., GORODNITSKY, I. F. et KUTAS, M. (2004). Automatic removal of eye movement and blink artifacts from EEG data using blind component separation. *Psychophysiology*, 41(2):313–325. 62
- [**Kachenoura et al., 2008**] KACHENOURA, A., ALBERA, L., SENHADJI, L. et COMON, P. (2008). Ica: a potential tool for bci systems. *Signal Processing Magazine, IEEE*, 25(1):57–68. 62
- [**Kaiser et al., 2001**] KAISER, J., PERELMOUTER, J., IVERSEN, I. H., NEUMANN, N., GHANAYIM, N., HINTERBERGER, T., KÜBLER, A., KOTCHOUBEY, B. et BIRBAUMER, N. (2001). Self-initiation of eeg-based communication in paralyzed patients. *Clinical Neurophysiology*, 112(3):551–554. 54

- [**Kamoussi et al., 2005**] KAMOUSI, B., LIU, Z. et HE, B. (2005). Classification of motor imagery tasks for brain-computer interface applications by means of two equivalent dipoles analysis. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 13(2):166–171. [63](#), [65](#), [143](#), [157](#)
- [**Kaper et al., 2004**] KAPER, M., MEINICKE, P., GROSEKATHOEFER, U., LINGNER, T. et RITTER, H. (2004). BCI competition 2003-data set Iib: support vector machines for the P300 speller paradigm. *Biomedical Engineering, IEEE Transactions on*, 51(6):1073–1076. [34](#), [78](#)
- [**Kohavi, 1995**] KOHAVI, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *In IJCAI*, pages 1137–1145. [165](#)
- [**Kohavi et John, 1997**] KOHAVI, R. et JOHN, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1–2):273–324. [153](#)
- [**Koles, 1991**] KOLES, Z. J. (1991). The quantitative extraction and topographic mapping of the abnormal components in the clinical eeg. *Electroencephalography and Clinical Neurophysiology*, 79(6):440–447. [59](#)
- [**Kybic et al., 2005**] KYBIC, J., CLERC, M., ABBOUD, T., FAUGERAS, O., KERIVEN, R. et PAPADOPOULO, T. (2005). A common formalism for the integral formulations of the forward eeg problem. *IEEE Transactions on Medical Imaging*, 24:12–28. [143](#)
- [**Lachaux et al., 2007**] LACHAUX, J.-P., JERBI, K., BERTRAND, O., MIOTTI, L., HOFFMANN, D., SCHOENDROFF, B. et KAHANE, P. (2007). Braintv: a novel approach for online mapping of human brain functions. *Biological Research*, 40:401–413. [25](#)
- [**Lal et al., 2004**] LAL, T., SCHRODER, M., HINTERBERGER, T., WESTON, J., BOGDAN, M., BIRBAUMER, N. et SCHOLKOPF, B. (2004). Support vector channel selection in BCI. *Biomedical Engineering, IEEE Transactions on*, 51(6):1003–1010. [76](#)
- [**Lalor et al., 2005**] LALOR, E. C., KELLY, S. P., FINUCANE, C., BURKE, R., SMITH, R., REILLY, R. B. et MCDARBY, G. (2005). Steady-state VEP-based brain-computer interface control in an immersive 3D gaming environment. *EURASIP J. Appl. Signal Process.*, 2005:3156–3164. [36](#)
- [**Le Pavec et Clerc, 2008**] LE PAVEC, J. et CLERC, M. (2008). Adaptive classification of mental states for asynchronous brain computer interfaces. *In Deuxième conférence française de Neurosciences Computationnelles*. [56](#), [117](#)
- [**Leahy et al., 1998**] LEAHY, R., MOSHER, J., SPENCER, M., HUANG, M. et LEWINE, J. (1998). A study of dipole localization accuracy for MEG and EEG using a human skull phantom. Los Alamos Technical Report LA-UR-98-1442, Los Alamos National Laboratory. Revision of LA-UR-97-4804. [26](#), [27](#)

- [**Lee et al., 1999**] LEE, T.-W., GIROLAMI, M. et SEJNOWSKI, T. J. (1999). Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources. *Neural Computation*. 61
- [**Leeb et al., 2007**] LEEB, R., LEE, F., KEINRATH, C., SCHERER, R., BISCHOF, H. et PFURTSCHELLER, G. (2007). Brain–computer communication: motivation, aim, and impact of exploring a virtual apartment. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 15(4):473–482. 18, 90
- [**Li et al., 2004**] LI, Y., GAO, X., LIU, H. et GAO, S. (2004). Classification of single-trial electroencephalogram during finger movement. *Biomedical Engineering, IEEE Transactions on*, 51(6):1019–1025. 76
- [**Long et al., 2004**] LONG, M. A., LANDISMAN, C. E. et CONNORS, B. W. (2004). Small clusters of electrically coupled neurons generate synchronous rhythms in the thalamic reticular nucleus. *The Journal of Neuroscience*, 24(2):341–349. 41
- [**Lopez-Larraz et al., 2010**] LOPEZ-LARRAZ, E., ITURRATE, I., MONTESANO, L. et MINGUEZ, J. (2010). Real-time recognition of feedback error-related potentials during a time-estimation task. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 2670–2673. 37, 38
- [**Lotte, 2008**] LOTTE, F. (2008). *Study of Electroencephalographic Signal Processing and Classification Techniques towards the use of Brain-Computer Interfaces in Virtual Reality Applications*. These, INSA de Rennes. 157
- [**Lotte et al., 2007**] LOTTE, F., CONGEDO, M., LÉCUYER, A., LAMARCHE, F. et ARNALDI, B. (2007). A review of classification algorithms for EEG-based brain-computer interfaces. *Journal of Neural Engineering*, 4:1–13. 78
- [**Lotte et al., 2008a**] LOTTE, F., LÉCUYER, A. et ARNALDI, B. (2008a). FURIA : un nouvel algorithme d'extraction de caractéristiques pour les interfaces cerveau-ordinateur utilisant modèles inverses et modèles flous. *Traitement du Signal*. 63, 64, 65, 143, 157, 158
- [**Lotte et al., 2008b**] LOTTE, F., RENARD, Y. et LÉCUYER, A. (2008b). Self-Paced Brain-Computer Interaction with Virtual Worlds: A Quantitative and Qualitative Study "Out of the Lab". In *4th international Brain Computer Interface Workshop and Training Course*, Graz, Autriche. Graz University of Technology. 55, 83, 84
- [**MacKay, 2005**] MACKEY, W. (2005). *Wheels of motion: oscillatory potentials in the motor cortex*, chapitre 7, pages 181–211. CRC Press, Boca Raton, FL. 41
- [**Mason et al., 2007**] MASON, S., BASHASHATI, A., FATOURECHI, M., NAVARRO, K. et BIRCH, G. (2007). A comprehensive survey of brain interface technology designs. *Annals of Biomedical Engineering*, 35(2):137–169. 14, 15, 19

- [**Mason et Birch, 2000**] MASON, S. et BIRCH, G. (2000). A brain-controlled switch for asynchronous control applications. *Biomedical Engineering, IEEE Transactions on*, 47(10):1297–1307. 52, 53, 55
- [**Mason et al., 2006**] MASON, S., KRONEGG, J., HUGGINS, J., FATOURECHI, M. et SCHLÖGL, A. (2006). Evaluating the performance of self-paced brain computer interface technology. Neil Squire Soc., Vancouver, BC, Canada, Tech. Rep. 19
- [**McFarland et al., 2006**] MCFARLAND, D., ANDERSON, C., MULLER, K.-R., SCHLOGL, A. et KRUSIENSKI, D. (2006). BCI meeting 2005-workshop on BCI signal processing: feature extraction and translation. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):135–138. 76
- [**McFarland et al., 1997a**] MCFARLAND, D., LEFKOWICZ, A. et WOLPAW, J. (1997a). Design and operation of an eeg-based brain-computer interface with digital signal processing technology. *Behavior Research Methods*, 29:337–345. 10.3758/BF03200585. 51
- [**McFarland et al., 1998**] MCFARLAND, D., MCCANE, L. et WOLPAW, J. (1998). EEG-based communication and control: Short-term role of feedback. *IEEE Transactions on Rehabilitation Engineering*, 6(1). 18
- [**McFarland et al., 1997b**] MCFARLAND, D. J., MCCANE, L. M., DAVID, S. V. et WOLPAW, J. R. (1997b). Spatial filter selection for eeg-based communication. *Electroencephalography and Clinical Neurophysiology*, 103(3):386–394. 52, 58
- [**McFarland et al., 2010**] MCFARLAND, D. J., SARNACKI, W. A. et WOLPAW, J. R. (2010). Electroencephalographic (eeg) control of three-dimensional movement. *Journal of Neural Engineering*, 7(3):036007. 44, 51, 69, 83
- [**McFarland et Wolpaw, 2008**] MCFARLAND, D. J. et WOLPAW, J. R. (2008). Sensorimotor rhythm-based brain-computer interface (bci): model order selection for autoregressive spectral analysis. *Journal of Neural Engineering*, 5(2):155. 52, 76
- [**Middendorff et al., 2000**] MIDDENDORF, M., MCMILLAN, G., CALHOUN, G. et JONES, K. (2000). Brain-computer interfaces based on the steady-state visual-evoked response. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):211–214. 54
- [**Millan et Mourino, 2003**] MILLAN, J. et MOURINO, J. (2003). Asynchronous bci and local neural classifiers: an overview of the adaptive brain interface project. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 11(2):159–161. 45, 53
- [**Morris et de Groot, 1981**] MORRIS et de GROOT, B. (1981). *Le Bandit manchot*, volume 79. Dargaud. 118

- [**Mosher et al., 1992**] MOSHER, J. C., LEWIS, P. S. et LEAHY, R. M. (1992). Multiple dipole modeling and localization from spatio-temporal meg data. *IEEE Transactions on Biomedical Engineering*, 39(6):541–553. 151
- [**Müller-Putz et al., 2010**] MÜLLER-PUTZ, G., KAISER, V., SOLIS-ESCALANTE, T. et PFURTSCHELLER, G. (2010). Fast set-up asynchronous brain-switch based on detection of foot motor imagery in 1-channel eeg. *Medical and Biological Engineering and Computing*, 48:229–233. 10.1007/s11517-009-0572-7. 83, 84
- [**Munssinger et al., 2010**] MUNSSINGER, J., HALDER, S., KLEIH, S., FURDEA, A., RACO, V., HOSLE, A. et KUBLER, A. (2010). Brain painting: First evaluation of a new brain-computer interface application with als-patients and healthy volunteers. *Frontiers in neuroscience*. 32, 49
- [**Naeem et al., 2006**] NAEEM, M., BRUNNER, C., LEEB, R., GRAIMANN, B. et PFURTSCHELLER, G. (2006). Separability of four-class motor imagery data using independent components analysis. *Journal of Neural Engineering*, 3(3):208. 62
- [**Nagaoka et al., 2010**] NAGAOKA, T., SAKATANI, K., AWANO, T., YOKOSE, N., HOSHINO, T., MURATA, Y., KATAYAMA, Y., ISHIKAWA, A. et EDA, H. (2010). Development of a new rehabilitation system based on a brain-computer interface using near-infrared spectroscopy. In TAKAHASHI, E. et BRULEY, D. F., éditeurs : *Oxygen Transport to Tissue XXXI*, volume 662 de *Advances in Experimental Medicine and Biology*, pages 497–503. Springer US. 29
- [**Nam et al., 2011**] NAM, C., JEON, Y., KIM, Y., LEE, I. et PARK, K. (2011). Movement imagery-related lateralization of event-related (de) synchronization (ERD/ERS): Motor-imagery duration effects. *Clinical Neurophysiology*, 122(3): 567–577. 92
- [**Noirhomme et al., 2008**] NOIRHOMME, Q., KITNEY, R. et MACQ, B. (2008). Single-trial EEG source reconstruction for brain-computer interface. *IEEE Trans. Biomed. Eng.*, 55(5):1592–1603. 64, 65, 143, 145, 157
- [**Nunez, 2005**] NUNEZ, P. (2005). *Electric fields of the brain: the neurophysics of EEG*. Oxford University Press. 26
- [**Olivi, 2011**] OLIVI, E. (2011). *Coupling of numerical methods for the forward problem in Magneto- and Electro-Encephalography*. Thèse de doctorat, Nice Sophia Antipolis University. 142
- [**Penfield et Boldrey, 1937**] PENFIELD, W. et BOLDREY, E. (1937). Somatic motor and sensory representation in the cerebral cortex of man as studied by electrical stimulation. *Brain*, 60(4):389–443. 42
- [**Penfield et Rasmussen, 1950**] PENFIELD, W. et RASMUSSEN, T. (1950). *The cerebral cortex of man*. New York: Macmillan. 43

- [**Pfurtscheller et al., 2010a**] PFURTSCHELLER, G., ALLISON, B. Z., BRUNNER, C., BAUERNFEIND, G., SOLIS-ESCALANTE, T., SCHERER, R., ZANDER, T. O., MUELLER-PUTZ, G., NEUPER, C. et BIRBAUMER, N. (2010a). The hybrid bci. *Frontiers Research Foundatio.* 49
- [**Pfurtscheller et Aranibar, 1979**] PFURTSCHELLER, G. et ARANIBAR, A. (1979). Evaluation of event-related desynchronization (ERD) preceding and following voluntary self-paced movement. *Electroencephalography and Clinical Neurophysiology*, 46(2):138 – 146. 97
- [**Pfurtscheller et al., 1996**] PFURTSCHELLER, G., JR., A. S. et NEUPER, C. (1996). Post-movement beta synchronization. a correlate of an idling motor area? *Electroencephalography and Clinical Neurophysiology*, 98(4):281 – 293. 83
- [**Pfurtscheller et Neuper, 2010**] PFURTSCHELLER, G. et NEUPER, C. (2010). Dynamics of sensorimotor oscillations in a motor task. In GRAIMANN, B., PFURTSCHELLER, G. et ALLISON, B., éditeurs : *Brain-Computer Interfaces*, The Frontiers Collection, pages 47–64. Springer Berlin Heidelberg. 100
- [**Pfurtscheller et al., 2005**] PFURTSCHELLER, G., NEUPER, C., BRUNNER, C. et da SILVA, F. L. (2005). Beta rebound after different types of motor imagery in man. *Neuroscience Letters*, 378(3):156 – 159. 56, 83, 94, 97
- [**Pfurtscheller et al., 1997**] PFURTSCHELLER, G., NEUPER, C., FLOTZINGER, D. et PREGENZER, M. (1997). EEG-based discrimination between imagination of right and left hand movement. *Electroenceph. Clin. Neurophysiol.*, 10:642–651. 76
- [**Pfurtscheller et al., 2000**] PFURTSCHELLER, G., NEUPER, C., GUGER, C., HARKAM, W., RAMOSER, H., SCHLOGL, A., OBERMAIER, B. et PREGENZER, M. (2000). Current trends in graz brain-computer interface (BCI) research. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):216 –219. 52
- [**Pfurtscheller et al., 2003**] PFURTSCHELLER, G., NEUPER, C., MULLER, G., OBERMAIER, B., KRAUSZ, G., SCHLOGL, A., SCHERER, R., GRAIMANN, B., KEINRATH, C., SKLIRIS, D., WORTZ, M., SUPP, G. et SCHRANK, C. (2003). Graz-BCI: state of the art and clinical applications. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 11(2):1 –4. 52
- [**Pfurtscheller et Solis-Escalante, 2009**] PFURTSCHELLER, G. et SOLIS-ESCALANTE, T. (2009). Could the beta rebound in the EEG be suitable to realize a “brain switch”? *Clinical Neurophysiology*, 120:24–29. 54, 83, 84, 92, 97
- [**Pfurtscheller et al., 2010b**] PFURTSCHELLER, G., SOLIS-ESCALANTE, T., ORTNER, R., LINORTNER, P. et MULLER-PUTZ, G. (2010b). Self-paced operation of an ssvp-based orthosis with and without an imagery-based ”brain

- switch:” a feasibility study towards a hybrid bci. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 18(4):409–414. 49, 53, 54
- [**Polich, 2007**] POLICH, J. (2007). Updating p300: An integrative theory of p3a and p3b. *Clinical Neurophysiology*, 118(10):2128–2148. 32
- [**Porro et al., 1996**] PORRO, C. A., FRANCESCATO, M. P., CETTOLO, V., DIAMOND, M. E., BARALDI, P., ZUIANI, C., BAZZOCCHI, M. et di PRAMPERO, P. E. (1996). Primary motor and sensory cortex activation during motor performance and motor imagery: A functional magnetic resonance imaging study. *The Journal of Neuroscience*, 16(23):7688–7698. 40
- [**Qian et al., 2010**] QIAN, K., NIKOLOV, P., HUANG, D., FEI, D.-Y., CHEN, X. et OU, B. (2010). A motor imagery-based online interactive brain-controlled switch: Paradigm development and preliminary test. *Clinical neurophysiology*. 54
- [**Qin et al., 2005**] QIN, J., LI, Y. et CICHOCKI, A. (2005). Ica and committee machine-based algorithm for cursor control in a bci system. In WANG, J., LIAO, X. et YI, Z., éditeurs : *Advances in Neural Networks – ISNN 2005*, volume 3496 de *Lecture Notes in Computer Science*, pages 293–318. Springer Berlin. 62
- [**Qin et al., 2004**] QIN, L., DING, L. et HE, B. (2004). Motor imagery classification by means of source analysis for brain–computer interface applications. *Journal of Neural Engineering*, 1(3):135. 63, 157
- [**Ramoser et al., 2000**] RAMOSER, H., MULLER-GERKING, J. et PFURTSCHELLER, G. (2000). Optimal spatial filtering of single trial eeg during imagined hand movement. *Rehabilitation Engineering, IEEE Transactions on*, 8(4):441–446. 59
- [**Renard et al., 2010**] RENARD, Y., LOTTE, F., GIBERT, G., CONGEDO, M., MABY, E., DELANNOY, V., BERTRAND, O. et LÉCUYER, A. (2010). Open-vibe: An open-source software platform to design, test, and use brain–computer interfaces in real and virtual environments. *Presence: Teleoperators and Virtual Environments*, 19(1):35–53. 8, 74, 163
- [**Rivet et al., 2009**] RIVET, B., SOULOUMIAC, A., ATTINA, V. et GIBERT, G. (2009). xdawn algorithm to enhance evoked potentials: Application to brain–computer interface. *Biomedical Engineering, IEEE Transactions on*, 56(8):2035–2043. 34
- [**Robbins, 1952**] ROBBINS, H. (1952). Some aspects of the sequential design of experiments. *Bulletin of the American Mathematics Society*, 58:527–535. 118, 120
- [**Sadeghian et Moradi, 2007**] SADEGHIAN, E. et MORADI, M. (2007). Continuous detection of motor imagery in a four-class asynchronous bci. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 3241–3244. 55

- [Schalk *et al.*, 2008] SCHALK, G., MILLER, K. J., ANDERSON, N. R., WILSON, J. A., SMYTH, M. D., OJEMANN, J. G., MORAN, D. W., WOLPAW, J. R. et LEUTHARDT, E. C. (2008). Two-dimensional movement control using electrocorticographic signals in humans. *Journal of Neural Engineering*, 5(1):75. 51, 52, 139
- [Scherer *et al.*, 2008] SCHERER, R., LEE, F., SCHLOGL, A., LEEB, R., BISCHOF, H. et PFURTSCHELLER, G. (2008). Toward self-paced brain-computer communication: navigation through virtual worlds. *Biomedical Engineering, IEEE Transactions on*, 55(2):675–682. 53, 57
- [Scherer *et al.*, 2007] SCHERER, R., MÜLLER-PUTZ, G. R. et PFURTSCHELLER, G. (2007). Self-initiation of eeg-based brain-computer communication using the heart rate response. *Journal of Neural Engineering*, 4(4):L23. 54
- [Schieber *et al.*, 1993] SCHIEBER, M. et HIBBARD, L. (1993). How somatotopic is the motor cortex hand area? *Science*, 261(5120):489–492. 42
- [Schloegl *et al.*, 1997] SCHLOEGL, A., LUGGER, K. et PFURTSCHELLER, G. (1997). Using adaptive autoregressive parameters for a brain-computer-interface experiment. In *Engineering in Medicine and Biology Society, 1997. Proceedings of the 19th Annual International Conference of the IEEE*, volume 4, pages 1533–1535 vol.4. 76
- [Schlögl *et al.*, 2005] SCHLÖGL, A., LEE, F., BISCHOF, H. et PFURTSCHELLER, G. (2005). Characterization of four-class motor imagery eeg data for the bci-competition 2005. *Journal of Neural Engineering*, 2(4):L14. 78
- [Sekihara *et al.*, 2002a] SEKIHARA, K., NAGARAJAN, S., POEPPPEL, D. et MARRANTZ, A. (2002a). Performance of an meg adaptive-beamformer technique in the presence of correlated neural activities: effects on signal intensity and time-course estimates. *Biomedical Engineering, IEEE Transactions on*, 49(12):1534–1546. 150
- [Sekihara *et al.*, 2002b] SEKIHARA, K., NAGARAJAN, S. S., POEPPPEL, D., MARRANTZ, A. et MIYASHITA, Y. (2002b). Application of an meg eigenspace beamformer to reconstructing spatio-temporal activities of neural sources. *Human Brain Mapping*, 15(4):199–215. 150, 151
- [Sellers *et al.*, 2006a] SELLERS, E., KUBLER, A. et DONCHIN, E. (2006a). Brain-computer interface research at the university of south florida cognitive psychophysiology laboratory: the p300 speller. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):221–224. 31
- [Sellers *et al.*, 2006] SELLERS, E. W. et DONCHIN, E. (2006). A p300-based brain-computer interface: Initial tests by als patients. *Clinical Neurophysiology*, 117(3):538–548. 32, 34
- [Sellers *et al.*, 2006b] SELLERS, E. W., KRUSIENSKI, D. J., MCFARLAND, D. J., VAUGHAN, T. M. et WOLPAW, J. R. (2006b). A p300 event-related potential

- brain-computer interface (bci): The effects of matrix size and inter stimulus interval on performance. *Biological Psychology*, 73(3):242 – 252. 34
- [Semlitsch *et al.*, 1986] SEMLITSCH, H. V., ANDERER, P., SCHUSTER, P. et PRESSLICH, O. (1986). A solution for reliable and valid reduction of ocular artifacts, applied to the p300 erp. *Psychophysiology*, 23(6):695–703. 16
- [Shannon et Weaver, 1964] SHANNON, C. E. et WEAVER, W. (1964). *The Mathematical Theory of Communication*. Urbana, IL: Univ. Illinois Press. 80
- [Shoker *et al.*, 2005] SHOKER, L., SANEI, S., WANG, W. et CHAMBERS, J. (2005). Removal of eye blinking artifact from the electro-encephalogram, incorporating a new constrained blind source separation algorithm. *Medical and Biological Engineering and Computing*, 43:290–295. 10.1007/BF02345968. 16
- [Spencer *et al.*, 1992] SPENCER, M., LEAHY, R., MOSHER, J. et LEWIS, P. (1992). Adaptive filters for monitoring localized brain activity from surface potential time series. In *Signals, Systems and Computers, 1992. 1992 Conference Record of The Twenty-Sixth Asilomar Conference on*, pages 156 –161 vol.1. 148
- [Strehl *et al.*, 2006] STREHL, U., LEINS, U., GOTH, G., KLINGER, C., HINTERBERGER, T. et BIRBAUMER, N. (November 2006). Self-regulation of slow cortical potentials: A new treatment for children with attention-deficit/hyperactivity disorder. *Pediatrics*, 118(5):e1530–e1540. 39, 40
- [Susila *et al.*, 2010] SUSILA, I. P., KANO, S., MIYAMOTO, K.-i. et YOSHINOBU, T. (2010). xbc: A generic platform for development of an online bci system. *IEEJ Transactions on Electrical and Electronic Engineering*, 5(4):467–473. 163
- [Takano *et al.*, 2009] TAKANO, K., KOMATSU, T., HATA, N., NAKAJIMA, Y. et KANSAKU, K. (2009). Visual stimuli for the p300 brain-computer interface: A comparison of white/gray and green/blue flicker matrices. *Clinical Neurophysiology*, 120(8):1562 – 1566. 34
- [Tangemann *et al.*, 2009] TANGERMANN, M. W., KRAULEDAT, M., GRZESKA, K., SAGEBAUM, M., BLANKERTZ, B., VIDAURRE, C. et MÜLLER, K.-R. (2009). Playing pinball with non-invasive BCI. *Advances in Neural Information Processing Systems*, 21:1641–1648. 42
- [Thomas *et al.*, 2012] THOMAS, E., FRUITET, J. et CLERC, M. (2012). Investigating brief motor imagery for an ERD/ERS based BCI. *EMBC 2012, IEEE Engineering in Medicine and Biology Conference*. 56, 57
- [Tibshirani, 1996] TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *J.R. Statist. Soc.*, 58(1):267–288. 153
- [Townsend *et al.*, 2004] TOWNSEND, G., GRAIMANN, B. et PFURTSCHELLER, G. (2004). Continuous eeg classification during motor imagery-simulation of an asynchronous bci. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 12(2):258 –265. 55

- [**Townsend et al., 2010**] TOWNSEND, G., LAPALLO, B., BOULAY, C., KRUSIENSKI, D., FRYE, G., HAUSER, C., SCHWARTZ, N., VAUGHAN, T., WOLPAW, J. et SELLERS, E. (2010). A novel p300-based brain-computer interface stimulus presentation paradigm: Moving beyond rows and columns. *Clinical Neurophysiology*, 121(7):1109 – 1120. 34
- [**Treder et Blankertz, 2010**] TREDER, M. et BLANKERTZ, B. (2010). (c)overt attention and visual speller design in an erp-based brain-computer interface. *Behavioral and Brain Functions*, 6(1):28. 34
- [**Van Veen, 1988**] VAN VEEN, B. (1988). Eigenstructure based partially adaptive array design. *Antennas and Propagation, IEEE Transactions on*, 36(3):357–362. 151
- [**Van Veen et Buckley, 1988**] VAN VEEN, B. et BUCKLEY, K. (1988). Beamforming: a versatile approach to spatial filtering. *IEEE Acoustics, Speech, Signal Processing Mag.*, 5:4–24. 146
- [**Van Veen et al., 1997**] VAN VEEN, B., VAN DRONGELEN, W., YUCHTMAN, M. et SUZUKI, A. (1997). Localization of brain electrical activity via linearly constrained minimum variance spatial filtering. *Biomedical Engineering, IEEE Transactions on*, 44(9):867–880. 150
- [**Vapnik, 1995**] VAPNIK, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag. 78
- [**Vapnik, 1998**] VAPNIK, V. (1998). *Statistical Learning Theory*. John Wiley. 78
- [**Vaughan et al., 2006**] VAUGHAN, T., MCFARLAND, D., SCHALK, G., SARNACKI, W., KRUSIENSKI, D., SELLERS, E. et WOLPAW, J. (2006). The Wadsworth BCI research and development program: at home with BCI. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):229–233. 18, 34
- [**Veen et al., 1997**] VEEN, B. D. V., van DRONGELEN, W., YUCHTMAN, M. et SUZUKI, A. (1997). Localization of brain electrical activity via linearly constrained minimum variance spatial filtering. *IEEE Transactions on Biomedical Engineering*, 44(9):867–880. 148
- [**Vidal, 1973**] VIDAL, J. J. (1973). Toward direct brain-computer communication. *Annual Review of Biophysics and Bioengineering*, 2(1):157–180. 7, 13
- [**Vidaurre et Blankertz, 2010**] VIDAURRE, C. et BLANKERTZ, B. (2010). Towards a cure for BCI illiteracy. *Brain Topography*, 23:194–198. 10.1007/s10548-009-0121-6. 109, 110, 117
- [**Vion-Dury et Blanquet, 2008**] VION-DURY, J. et BLANQUET, F. (2008). *Pratique de l'EEG*. Masson. 26
- [**Weiskopf et al., 2004**] WEISKOPF, N., MATHIAK, K., BOCK, S., SCHARNOWSKI, F., VEIT, R., GRODD, W., GOEBEL, R. et BIRBAUMER, N. (2004).

- Principles of a brain-computer interface (BCI) based on real-time functional magnetic resonance imaging (fMRI). *Biomedical Engineering, IEEE Transactions on*, 51(6):966–970. 29
- [**Welch, 1967**] WELCH, P. (1967). The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *Audio and Electroacoustics, IEEE Transactions on*, 15(2):70–73. 76
- [**Wolpaw, 2007**] WOLPAW, J. (2007). Brain-computer interfaces as new brain output pathways. *J. Physiol.*, 579(3):613–619. 17, 18, 45
- [**Wolpaw et al., 2002**] WOLPAW, J., BIRBAUMER, N., MCFARLAND, D., PFURTSCHELLER, G. et VAUGHAN, T. (2002). Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113:767–791. 45, 52
- [**Wolpaw et al., 2006**] WOLPAW, J., LOEB, G., ALLISON, B., DONCHIN, E., do NASCIMENTO, O., HEETDERKS, W., NIJBOER, F., SHAIN, W. et TURNER, J. (2006). BCI meeting 2005-workshop on signals and recording methods. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):138–141. 23
- [**Wolpaw et McFarland, 2004**] WOLPAW, J. et MCFARLAND, D. (2004). Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *PNAS*, 101(51):17849–17854. 44, 51, 69
- [**Wolpaw et al., 2000**] WOLPAW, J. R., BIRBAUMER, N., HEETDERKS, W. J., MCFARLAND, D. J., PECKHAM, P. H., SCHALK, G., DONCHIN, E., QUATRANO, L. A., ROBINSON, C. J. et VAUGHAN, T. M. (2000). Brain-computer interface technology: a review of the first international meeting. *IEEE transactions on rehabilitation engineering a publication of the IEEE Engineering in Medicine and Biology Society*, 8(2):164–73. 80, 83
- [**Wolpaw et al., 1991**] WOLPAW, J. R., MCFARLAND, D. J., NEAT, G. W. et FORNERIS, C. A. (1991). An eeg-based brain-computer interface for cursor control. *Electroencephalography and Clinical Neurophysiology*, 78(3):252–259. 51, 69
- [**Wu et al., 2009**] WU, W., CHEN, Z., GAO, S. et BROWN, E. (2009). A probabilistic framework for learning robust common spatial patterns. *In Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 4658–4661. 61
- [**Yan et al., 2008**] YAN, T., JINGTIAN, T. et ANDONG, G. (2008). Multi-class eeg classification for brain computer interface based on csp. *In BioMedical Engineering and Informatics, 2008. BMEI 2008. International Conference on*, volume 2, pages 469–472. 60

-
- [[Yoo et al., 2004](#)] YOO, S.-S., FAIRNENY, T., CHEN, N.-K., CHOO, S.-E., PANNYCH, L. P., PARK, H., LEE, S.-Y. et JOLESZ, F. A. (2004). Brain-computer interface using fmri: spatial navigation by thoughts. *NeuroReport*, 15. [29](#)
- [[Zhang et al., 2007](#)] ZHANG, D., WANG, Y., GAO, X., HONG, B. et GAO, S. (2007). An algorithm for idle-state detection in motor-imagery-based brain-computer interface. *Intell. Neuroscience*, 2007:5–5. [55](#)