

Motor Intention in the Posterior Parietal Cortex - Experimental Data Analysis and Functional Modeling Study

THÈSE N° 4253 (2008)

PRÉSENTÉE LE 12 DÉCEMBRE 2008

À LA FACULTE INFORMATIQUE ET COMMUNICATIONS

LABORATOIRE DE SYSTÈMES NON LINÉAIRES

PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Jugoslava AĆIMOVIĆ

acceptée sur proposition du jury:

Prof. J.-Y. Le Boudec, président du jury

Prof. M. Hasler, directeur de thèse

Prof. W. Gerstner, rapporteur

Prof. G. Innocenti, rapporteur

Dr M.-L. Linne, rapporteur



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2008

Abstract

The complexity of processes occurring in the brain is an intriguing issue not just for scientists and medical doctors, but the humanity in general. The cortex ability to perceive and analyze an enormous amount of information in an instance of time, the parallelism and computational efficiency are among the questions that attract attention. Even a simple, everyday gesture, for example, reaching for a cup of coffee, evokes a flow of signals in the brain. It goes from the primary visual region, that locates the cup on the table, to the primary motor region that sends the precise coordinates to the hand, and the instruction what to do next. The sequence of signal transmission and transformation continues through several regions, sensory, associative and motor ones. In this study, we will focus on the posterior parietal cortex, the region involved in the transformation of visual inputs into the preliminary motor plans.

The years of experimental work revealed mechanisms for integration of multimodal signals, coordinate transformations, information representation in multiple coordinate frames, and many other. Still, a single encompassing theory about movement generation in the parietal cortex does not exist, and is a matter of debate. This study contributes to the analysis of motor intention in the 7a parietal region. The motor intention, a high-level cognitive signal, is defined as the preliminary plan for making a movement. From the engineering point of view, encoding of motor parameters in the neural activity is extensively studied within the framework of brain-computer interfaces. The motivation behind these studies is the development of neural prosthesis for the paralyzed persons. The direct cortical prosthesis can significantly improve the lives of paralyzed people, who have lost every other contact with the outside world. Also, this framework opens the possibilities for monitoring the neural processes during the execution of natural movements, and studying the mechanisms behind it.

In this work, a method for identification of motor intention from the standard recordings of neural activity, the spike trains, is developed. The data of interest was collected in a series of behavioral experiments involving reaching or saccadic eye movements. The presence and absence of motor intention was monitored in various phases of motion execution, and for different types of movements. All the recordings obtained simultaneously are combined in the same decoding session. Therefore, the analysis is done using the activity of small population of cells (typically 8 to 12 cells). We aim to study the motor intention in a general context which requires using activity of multiple cells. The population size is determined by the experimental procedure. Throughout this study we assume that the motor intention can be read from the spike rates, the assumption supported by the neurophysiological studies. Therefore, all the simultaneously collected spike trains are converted into vectors of spike rates. The results of this study show that motor intention can be decoded from the spike rates. A machine-learning based algorithm is developed to analyze the presence or absence of motor intention in the obtained spike rate vectors. This algorithm, based on standard support vector machines, can distinguish between the segments of recordings that encode motor intention, from those that do not encode it. The goal of the study was to examine the precision of motor intention identification, when the activity of a randomly selected set of cells is analyzed using on such algorithm. Additionally, several relevant parameters were tested. The algorithm precision during different phases of movement execution is tested. Also, the influence of the population size and of the procedure for spike rates computation is examined.

The obtained results demonstrated that the motor intention can be extracted from the neural signals with the precision of around 70% for a randomly selected set of cells. For the best groups of cells, this precision was 82%. The motor intention identification was particularly precise during the intervals of preparation and realization of saccadic eye movements. This is in accordance with the known functions of the 7a region, where the majority of cells respond to the eye movements. The algorithm precision is determined by the considered population size. For the bigger population the precision increases. Still, this conclusion holds only on average, since adding one or a couple of randomly selected cells does not have to change the result. Randomly selected cells do not necessary carry the information of interest. The influence of each of the cells, present in one set, is tested in this context. The obtained results indicate redundant coding of motor intention in the parietal cortex. Many cells carry the same information, and some of them can be removed from the set without changing the algorithm precision. Still, removing all of them degrades the result. Finally, the

influence of the window size, used to compute spike rates in some of the tests is studied. In general, the precision improves when using bigger windows, the result that is consistent with the literature. Introducing the window for computing spike rates enables automatic identification of motor intention, the method suitable for the brain-computer interface applications.

Finally, the analysis of the experimental data is complemented with the study of an appropriately designed model. Modeling the biological processes, in order to reveal additional functionality and test some parameters not accessible through the data, is a widely accepted approach. Still, the development of a model, sufficiently simple for implementation on the standard hardware, sufficiently tractable in the simulations, yet informative enough to capture the main processes of interest, is not straightforward. Our motivation for accepting this approach was to test several parameters that imposed themselves as important in the data analysis step. Due to the nature of the problem itself, the test on an approximative model was the only feasible tactic. The influence of the population size and the window size was assessed in this study. This, additionally, demonstrated the algorithm precision scaling as a function of the number of cells.

Keywords: area 7a, bio-inspired models, brain-computer interface, hand-eyes coordination, liquid state machines, machine learning, motor intention, motor parameters, neural signals, population coding, posterior parietal cortex, recurrent networks, spike trains, spike rates, support vector machines.

Version abrégée

La complexité des processus se déroulant dans le cerveau est une intrigue pas seulement pour les scientifiques et les docteurs, mais pour l'humanité en général. L'habilité du cortex à percevoir et analyser une énorme quantité d'information en une fraction de temps, son efficacité et son architecture parallèle sont questions d'intérêt particulier. Même un simple geste de tous les jours, par exemple atteindre une tasse de café, génère un flux de signaux dans le cerveau, partant de la région visuelle primaire qui localise la tasse de café sur la table jusqu'à la région moteur primaire qui envoie les coordonnées précises à la main et les instructions de la tâche à exécuter. La séquence de transmissions et de transformations des signaux se déroule à travers plusieurs régions sensorielles, associatives et finalement à la région motrice. Dans cette étude, nous allons nous focaliser sur le cortex pariétal postérieur, une région impliquée dans la transformation des informations visuelles vers les plans préliminaires du mouvement.

Des années de travail expérimental ont révélé des mécanismes pour l'intégration des signaux multimodaux, la transformation de coordonnées, la représentation des informations dans des coordonnées multiples ainsi que beaucoup d'autres mécanismes. Néanmoins, une théorie unique comprenant la génération du mouvement dans le cortex pariétal n'existe pas et est sujette à discussion. Cette étude contribue à l'analyse des paramètres moteurs dans le cortex pariétal, particulièrement dans la région 7a. La notion d'intention de mouvement, un signal cognitif complexe, est définie comme la planification préliminaire d'un mouvement. D'un point de vue d'ingénieur, le codage des paramètres moteurs de l'activité neuronale est intensément étudié dans le cadre de l'interface cerveau-machine. La motivation derrière ces études est le développement de prothèses neurologiques pour les personnes paralysées. La prothèse corticale directe peut fortement améliorer la vie des personnes paralysées qui ont perdu tout autre contact avec le monde extérieur. Ce cadre de recherche ouvre aussi la possibilité d'enregistrer les processus neuronaux durant l'exécution de mouvements naturels et d'étudier les mécanismes sous-jacents.

Dans ce travail, une méthode pour l'identification de l'intention motrice à partir d'enregistrements standards d'activité neuronale (trains d'impulsions) est développée. Les données ont été collectées à partir d'une série d'expériences comportementalistes impliquant des mouvements oculaires et des mouvements pour atteindre un objet. La présence et l'absence d'intentions motrices ont été mesurées dans diverses phases de l'exécution du mouvement et pour différents types de mouvements. Ainsi l'analyse est faite en utilisant l'activité d'une petite population de cellules (typiquement de 8 à 12). Nous souhaitons étudier l'intention motrice dans un contexte global qui nécessite l'utilisation de l'activité de plusieurs cellules. La taille de la population est déterminée par la procédure expérimentale. A travers cette étude, nous supposons que l'intention motrice peut être lue à partir de la fréquence des impulsions, une hypothèse soutenue par les études neurophysiologiques. C'est pourquoi tous les trains d'impulsions collectés simultanément sont convertis en un vecteur de fréquence d'impulsions. Les résultats de cette étude montrent que l'intention motrice peut être décodée à partir de la fréquence des impulsions. Un algorithme basé sur l'apprentissage automatique est développé pour analyser la présence ou l'absence d'intention motrice dans le vecteur de fréquence d'impulsions. Cet algorithme peut différencier les périodes d'enregistrements qui encodent une intention motrice et ceux qui n'en encodent pas. Le but de l'étude est d'examiner la précision de l'identification d'une intention motrice quand l'activité d'un groupe aléatoire de cellules est analysée avec cet algorithme. De plus, la précision de l'algorithme est testée durant différentes phases du mouvement. Finalement la taille de la population et la procédure de calcul des impulsions sont examinées.

Les résultats obtenus démontrent que l'intention motrice peut être extraite des signaux neuronaux avec une précision d'environ 70% pour un groupe de cellules sélectionnées aléatoirement. Pour les meilleurs groupes de cellules, la précision atteint 82%. L'identification d'intentions motrices est particulièrement précise pendant les phases de préparation et la réalisation de mouvements oculaires. Cela est cohérent avec la fonction connue de la région 7a, où la majorité des cellules répondent aux mouvements oculaires. La précision est déterminée par la taille de la population considérée : la précision augmente avec une plus grande population. Néanmoins cette conclusion n'est valable qu'en général : ajouter une ou plusieurs cellules sélectionnées aléatoirement ne change pas nécessairement le résultat, car ces cellules peuvent ne pas apporter d'informations supplémentaires. L'influence de

chaque cellule présente dans un groupe est testée dans ce contexte. Les résultats obtenus montrent que le codage de l'intention motrice dans le cortex pariétal est redondant. Plusieurs cellules portent la même information et certaines d'entre elles peuvent être retirée du groupe sans que la précision de l'algorithme change. Néanmoins toutes les retirer dégrade le résultat. Finalement, l'influence de la taille de la fenêtre utilisée pour le calcul des impulsions est étudiée. En général, la précision augmente lorsque l'on utilise des fenêtres plus larges, résultat cohérent avec la littérature.

Finalement, l'analyse des données expérimentales est complétée par une étude d'un modèle approprié. Modéliser des processus biologiques, afin de trouver des fonctions additionnelles et tester des paramètres non-accessibles à travers les données, est une approche largement acceptée. Néanmoins, le développement d'un modèle, suffisamment simple pour être implémenté sur les machines actuelles, mais suffisamment complexe pour capturer les processus d'intérêts, n'est pas évident. Notre motivation pour valider cette approche a été de tester plusieurs paramètres qui s'imposent comme importants dans les étapes d'analyse de données. Dû aux limitations des expériences, ainsi qu'à la nature du problème, les tests sur le modèle, même approximatif, étaient la seule tactique possible. L'influence de la taille de la population et le taille de la fenêtre a été caractérisée dans cette étude. Cela démontre que la précision de l'algorithme est fonction du nombre de cellules.

Mots-clés: apprentissage automatique, intention motrice, codage de population, coordination main-oeil, cortex pariétal postérieur, fréquence d'impulsions, interfaces cerveau-machine, machines à état liquide, modèles bio-inspirés, paramètres moteurs, région 7a, réseaux récurrents, signal neurologique, train d'impulsions.

Acknowledgments

During my doctoral studies I was fortunate to meet many extraordinary people whose support and understanding helped me to successfully conclude my project.

I would like to express gratitude to my thesis adviser, Prof. Martin Hasler, for his support and patience during the four years that I spent in this laboratory. Many things that I learn about science and life will follow me for all my life.

Also, I would like to thank to my collaborators on this project, Prof. Alexandra Battaglia-Mayer and Prof. Roberto Caminiti from the University of Rome, whose experimental work on the parietal cortex functions served as the inspiration for this whole project. I am very grateful for their numerous explanations of the experiments, data, and in general, neurophysiologic aspects of this project, and also for their hospitality during my visits to Rome.

Mr. Alessandro Palladini, from the University of Bologna, had an important contribution to development of the artificial model presented here. His interest in this project, and his positive and dynamic approach to the work, in general, made this part of the project possible.

I had a great pleasure to spend some time outside of EPFL, on the internships in Belgium and Finland, during the second and the fourth year of my doctoral studies. These travels gave me the opportunity to look at my work from the new perspectives, but also, to learn a lot about other countries and cultures. I would like to thank to Prof. Johan Suykens from KU Leuven, and Dr. Marja-Leena Linne and Prof. Olli Yli-Harja from Tampere University of Technology, for their hospitality and help.

Finally, I would like to thank to the members of my thesis committee, Prof. Jean-Yves Le Boudec, the president of the jury, and Dr. Marja-Leena Linne, Prof. Giorgio Innocenti and Prof. Wulfram Gerstner, the committee members. Thank you for reading and correcting my thesis, and for the very interesting discussion of the results during my defense.

I really appreciated the atmosphere in the laboratory and the communication among lab members. It was great to work in a friendly and supportive environment, which I will always remember. Particularly, I would like to thank to Kumiko, who shared the office with me during all these years, and whose patience and calmness helped me to deal better with all the challenges of doctoral studies.

I would like to thank to my parents and my brother for their love and care, and their unconditional support and understanding in many difficult moments of my life. I know how difficult for you was all this time that we spent far from each others.

I was very fortunate to meet many wonderful people since coming to Lausanne, and make some friendships for life. If every new friend has made me a little bit better person, then, I have already gained a lot from coming to Lausanne. Many thanks to Zorana, my best friend and roommate during the last three years.

And Stane, I actually do not know how to thank you. I do not think it is possible to express it in words, how thankful I am for having you in my life, how important you are to me. I would not manage to finish this without you.

Contents

Acknowledgments	vii
1 Introduction	1
2 Neurophysiological background and motivation: The functional role of the posterior parietal cortex.	3
2.1 The mammalian neocortex	3
2.2 The posterior parietal cortex	5
2.2.1 The role of the posterior parietal cortex - studies on patients with parietal lesions	5
2.2.2 Regions of the posterior parietal cortex	6
Architecture of the PPC	6
Functional roles of the PPC regions	9
2.2.3 Electrophysiological studies	12
Preferred direction and population coding	12
Spatial information representation and coordinate transformations	13
The proposed models of the hand-eyes coordination in the PPC	16
Intention coding	16
2.3 Behavioral Experiments	17
2.3.1 Measurement apparatus	17
2.3.2 The position of the electrodes	19
2.3.3 The experiments	20
2.4 Database description	24
2.5 Cell types and information coding in the 7a parietal region	24
2.6 Motivation	28
3 Machine learning based approach to motor intention identification from the experimental data.	33
3.1 Problem statement	33
3.2 Overview of brain-computer interface studies	36
3.2.1 Noninvasive BCI and other alternatives to spike trains	38
3.2.2 Recording spike trains for the BCI applications	40
3.2.3 Algorithms, functions, and precision	41
3.2.4 State estimation for the autonomous BCI	46
3.3 Data Analysis Methods	49
3.3.1 Two Approaches to Feature Extraction From Spike Trains	49
Epochs classification	51
Single cell influence	53
Sliding window classification	54

3.4	Binary Classification Using Support Vector Machines	57
3.4.1	Support Vector Machines Algorithm	58
3.4.2	Implementation details	61
4	Experimental data study	65
4.1	Epochs classification results	65
4.2	The error distribution over experimental epochs	67
4.3	Single cell influence in a small population	71
4.4	Sliding window classification results	73
4.5	Discussion: Methods comparison and relevant algorithm parameters	79
5	Artificial neural network	85
5.1	Background: RNN and neuronal network models	86
5.2	Building elements	87
5.2.1	LIF neuron model	87
5.2.2	Synapse model	88
5.2.3	Network	90
5.3	Formalization of the behavioral tasks	90
5.4	Model and implementation details	92
5.4.1	Input layer	92
5.4.2	Hidden layer/RNN	93
5.4.3	Output	93
5.5	Tuning and selection of networks	94
5.5.1	Randomly generated networks and selection criteria	94
5.5.2	Training phase	95
5.6	Tasks classification	96
5.6.1	Binary classification	97
5.6.2	Results: Classification precision as a function of population size	98
5.7	Conclusion	101
6	Conclusion	103
A	Spiking network: model and parameters tuning	105
A.1	Input signals	105
A.2	Input layer parameters	105
A.3	Hidden layer / Recurrent neural network	106
A.4	Linear output	107
A.5	Parameters selection	107
	Bibliography	109
A.6	CV	116

Chapter 1

Introduction

The motivation behind this work is a larger study examining functions of the posterior parietal cortex, particularly the mechanisms for hand-eyes coordination. In a series of recording sessions a database, the starting point for this study, was collected. Two macaque monkeys were trained to perform a series of behavioral tasks, some of them involving a reaching movement, other a saccadic eye movement, or the coordinated movement of both, or no movement at all. A complex set of tasks was designed to capture the influence of as much as possible different signals occurring in the parietal cortex during a typical reaching movement. The activity of a large population of cells was recorded in the parietal area 7a. The detailed description of the behavioral experiments, the collected data set, and the conclusions obtained from that data are presented in [9, 10]. This study gave an important contribution to understanding the processes in the parietal cortex related to initialization of a movement, processing of sensory inputs, and relating them to the motor outputs.

In our study we focused on identification of one particular information, the motor intention. This high-level cognitive signal is also defined as a preliminary plan for a movement. The considered region is suitable for accessing such type of signals, since it represents a multimodal association area, where the sensory inputs evoke motor commands sent further to the motor regions. Previous studies of motor intention [4] demonstrated the existence of this signal and its influence on the cell activity modulation. A set of appropriate tasks was developed in order to reveal the information of interest. Here, we aim to identify the same signal under a more general condition. The goal of the proposed data analysis approach was to extract the motor intention along the neural recordings, whenever preset. The description of the methods and the obtained conclusions are presented in details in this study. The achievable precision of such a task is estimated, and the influence of several relevant parameters is assessed. Particularly important are the number of cells involved in the motor intention decoding task, and the integration time used to evaluate average cell activity. Also, the precision of signal estimation under various experimental conditions was assessed. The ultimate goal of this analysis is to develop a method that can detect the motor intention in time, for any type of movement, under any experimental conditions.

A study of motor parameters in the cortex is inevitably related to the field of brain-computer interface. The aim of this field, in general, is to provide an automatic system that can read neural signals and translate them into a set of commands, either for a cursor, or robot arm, or any other similar device [16, 83, 77]. Building a system directly controlled by human mind is not just a challenging problem. A reliable and long-term stable BCI can provide additional quality in the life of severely paralyzed persons, that cannot easily communicate with the outside world. Many efforts are directed toward development of non-invasive BCI, that can be used, for example, during the rehabilitation of patients that partially lost their motor abilities. Alternatively, they can find their application also in the augmentation of the

normal body functions. The typical BCI studies assess the reliability of the recording methods, develop the algorithms for automatic generation of motor control signals, and test the influence of various artificial extensions that can be controlled by such signals. Another aspect of these studies is theoretical, they provide a framework for studying motor signals directly obtained during a movement execution. The algorithms used here represent a new approach to data analysis, compared to the standard statistical methods. The motor intention, the focus of our study, can be seen as an additional control signal for such devices, that indicates a precise timing for the planned movements.

In addition to the data study, we tested the influence of relevant parameters on the approximative models. Combining the model studies with the data analysis can provide additional perspective on the considered problem. The presented model is far from a biological reality, and mimic just some selected features of the system of interest. Some reports demonstrated that even simple, appropriately chosen models can be used to reproduce behaviors observed in the biological systems and to analyze the mechanisms behind them [75, 102]. Our study on the model, supported the conclusion observed from the data analysis, that cannot be assessed otherwise due to the experimentally and methodologically imposed limitations.

We implemented a spiking recurrent neural network with input and output protocol that imitates some of the tasks used in the experiments from [9]. The motivation behind a particular networks choice is related to the tested parameters. We wanted to evaluate the change of the algorithm precision as a function of the number of the available cells, therefore we needed to develop a network as big as possible with respect to the computational facilities. Also, we tested the question of integration time, which required the use of a spiking network exhibiting a certain memory.

This text is organized as follows. In the first chapter the biological background is presented, the anatomical organization of the parietal cortex, as well as the functionality of its distinct regions is reviewed. Some of its known functional properties are also listed, for example, the gain field effect, the coordinate transform, and the population coding of directions. Also, the chapter presents some key studies showing the existence of motor intention. In the continuation of the chapter, the experimental set up is described in details, the measurement apparatus as well as the set of performed tasks. This is followed by an overview of the results obtained from the same database, and presented in [9, 10]. Finally, the motivation behind this study is elaborated at the end of the chapter.

The second chapter places the problem of interest in the engineering context. In the first section, the question of motor intention identification is formulated as a binary classification problem. It is followed by the overview of the BCI methods, results and challenges is given at the beginning of the chapter. In the second half of the chapter the detailed description of the implemented data analysis method is presented.

The third chapter is, probably, the key one in this text. There, the detailed description of the obtained results is given, together with the elaboration of their meaning and reasons behind them. As already stated, the particular attention is directed toward the three relevant parameters. The detailed analysis of the experimental epochs and their relation to motor intention in the view of the obtained results is presented. Then, the influence of the the population size and integration time used for calculating spike rates is assessed.

In the fourth chapter, the approximative model, based on a spiking recurrent network exposed to the input-output protocol similar to some of the experiments is presented. Some of the relevant parameters discussed in the Chapter 3 are revisited here, in the new context.

The manuscript is ending with the overview of the obtained conclusion and the potential future developments.

Chapter 2

Neurophysiological background and motivation: The functional role of the posterior parietal cortex.

A brief overview of the neurophysiological background is presented in this chapter. The first section lists some well known facts about the neocortex architecture and information processing mechanisms, and introduces the necessary terminology. The continuation of the chapter gives a literature overview of the posterior parietal cortex studies, and the main conclusions obtained from these studies. Finally, the experimental setup and the related data analysis, the motivation for the work presented in this manuscript, are described at the end of the chapter.

2.1 The mammalian neocortex

The rough anatomic division of the primate neocortex defines the six, anatomically and functionally distinct, regions. Frontal, temporal, occipital and parietal lobes are visible on the lateral surface of the neocortex, while cingulate cortex occupies the medial surface surrounding the corpus callosum, the set of fibers connecting the correlated regions in the opposite hemispheres. Finally, the insular cortex occupies the region that separates the temporal, from the parietal and frontal cortex, and is effectively hidden by the overgrowth of these three lobes. Figure 2.1 shows the lateral image of the left hemisphere of the human brain, together with the borders between the four cortical regions visible from this perspective. An important feature of the primate cortex is its specific shape, composed of alternating elevated convex and depressed concave regions, called gyri and sulci, respectively. Three particularly pronounced sulci are used to define the borders between the cortex regions - the lateral sulcus (the sylvian fissure) separates the temporal from the parietal and frontal regions, the central sulcus separates the frontal from the parietal lobe, and the parieto-occipital sulcus separates the parietal from the occipital lobe [44].

The parcellation shown on the Figure 2.1 refers to the widely accepted anatomy-based definition of the cortex areas, proposed by K. Brodmann in 1909. In order to give an informal description of these concepts, it is necessary to introduce some basic notions related to the cortex architecture, i.e. layers and columns. The thickness of the cortex is 2-4mm for most of the species, and it can be divided into six layers according to the typical cell types present. In general, the same structure is observed in most of the cortex regions, but the variations in thickness of certain layers, and in intensity of fibers vary a lot in certain regions, which enables their identification.

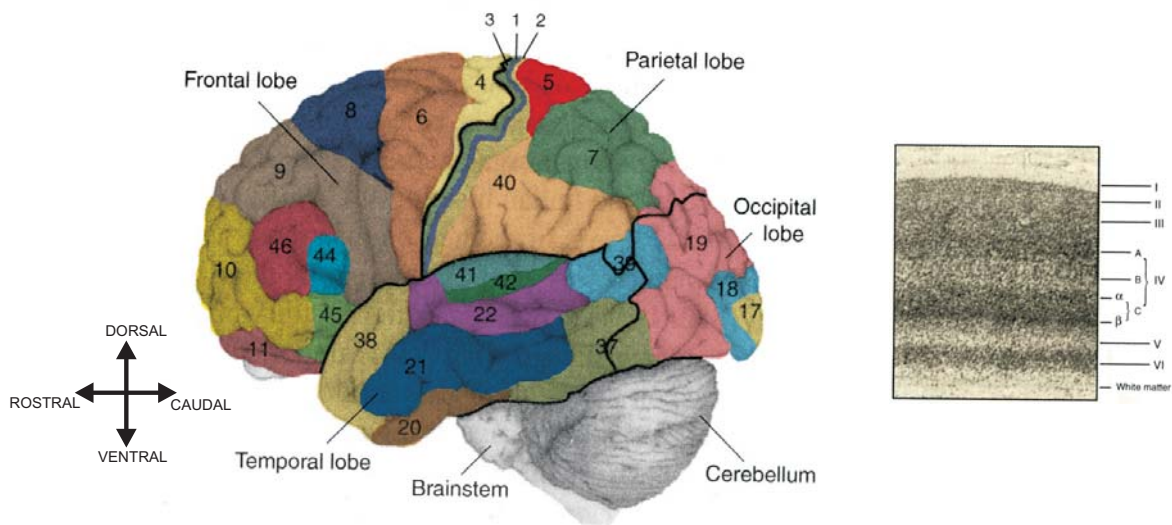


Figure 2.1: Left panel: lateral view of the left hemisphere of the human brain; the four lobes, together with the cerebellum and brainstem are shown. The visible Brodmann areas are numerated and color coded. Right panel: an example of cortical column, six layers and subdivisions of the fourth layer are shown. Adapted from T. P. Trappenberg: *Fundamentals of Computational Neuroscience*, Oxford University Press (2002). [94]

Figure 2.1 shows the horizontal cross section of the cortex divided into six layers, and the four subdivisions of the fourth layer. According to [44], the first one consists mainly of fibers spreading across regions, practically without cell bodies. The second and the fourth layers contain the smaller granular cells, that contribute to the local connections mainly. The third and the fifth layer consists of pyramidal cells, the most common type in the neocortex representing 75% to 90% of all the cells. The sixth layer is more inhomogeneous and contains various cell types, atypical for the other layers. It blends into the white matter, the region filled with the axon bundles forming the cortico-cortical connections. Within small areas of 300 to 600 μm in diameter, cells exhibit pronounced vertical connections across the layers compared to the lateral connections within the same layer, forming the columns, the elementary processing units in the cortex [44, 61].

The original Brodmann's parcellation into 47 distinct regions is based on the observed variations in the structure of cortical columns, the present cell types and the thickness of certain layers. The progress in studies of cortex anatomy, as well as gaining knowledge on functional roles of certain regions, introduced additional regions in this scheme.

Alternative cortex study is based on identification of functional regions, for example, according to their role in sensory information processing and generation of motor commands. This process is hierarchical, the information received through periphery sensory cells is transmitted to the primary sensory regions, then further to the unimodal association areas that integrate information related to one single type of sensations, e.g. vision, auditory sensations etc. The next step in information processing happens in the multimodal associative areas which integrate sensations of different modalities, in order to extract more complex information and generate output signals - proper reactions to the received inputs. These signals are further conveyed to the primary motor association areas responsible for initial planning of motor commands. Finally the premotor and primary motor cortex directly control movements, following the received inputs.

This study is focused on posterior parietal cortex and the overview of its anatomic and

functional properties is given in the following section.

2.2 The posterior parietal cortex

The parietal cortex (PC), shown on the Figure 2.1, is anatomically defined by the positions of the three pronounced sulci; the lateral sulcus separates it from the temporal lobe, the central sulcus from the frontal lobe, and the parieto-occipital sulcus from the occipital lobe. Brodmann areas 1,2, and 3, placed on the postcentral gyrus, are the most anterior part of the parietal cortex, and represent the primary somatosensory region. The posterior parietal cortex (PPC) in humans extends to the supramarginal and angular gyrus, the regions classified as Brodmann area 40 and 39, respectively. The latter occupies the junction of the parietal, temporal and occipital lobes. These two areas are involved in language processing, spatial orientation and semantic representation. They are described only in the context of studies in humans, since no evidence of the equivalent regions is found in monkeys [62].

The regions of interest for this study are the Brodmann areas 5 and 7, the surface inside the intraparietal sulcus (IPS), and the related regions on the medial wall of the hemisphere. In humans, both regions 5 and 7 belong to the superior parietal lobe (SPL), while the inferior parietal lobe (IPL) refers to the Brodmann areas 39 and 40. The anatomical organization of the monkey cortex is somewhat different and SPL contains the area 5, while the area 7 belongs to the IPL. Since most of the presented conclusions come from the studies on monkeys, the terms PPC, SPL and IPL will be mostly used in the context of the monkey cortex. Their anatomy and functional roles will be described in more details in what follows.

2.2.1 The role of the posterior parietal cortex - studies on patients with parietal lesions

Valuable knowledge on the PPC functions has been collected through clinical examination of patients with parietal lesions. Equivalent studies are carried on monkeys after chirurgical removal of certain regions in PPC, in order to monitor the consequences. Most of the literature on functional roles and processes in PPC lists some typical deficits in behavior observed in patients; for example [62] gives one detailed overview, while [7] aims to relate the deficits with the functional roles of the missing regions in the PPC, and draws conclusions in accordance with some recent discoveries of computational mechanisms in the PPC.

The examined human subjects suffered from lesions of different size and position within the PPC, while the visual and motor regions remained intact. Typically, the patients had no significant visual, somatosensory or motor disorders, but they demonstrated deficits in reaching and grasping due to the inability to connect sensory and motor functions. These observations emphasized the role of PPC as a multimodal associative area, integrating sensory inputs of different modalities, and initiating motor commands, later developed in the premotor and motor areas. Some common disorders reported in the literature are optic ataxia, together with different forms of apraxias and neglects.

The term optic ataxia is first introduced by Balint (1909), and is characterized by incapacity to properly complete visually guided reaching and grasping tasks. A specific property of this behavioral disorder is the hemispheric asymmetry. Right-handed patients with unilateral lesions on the right hemisphere, exhibited the 'visual field effect', the deficit in reaching for both hands in the contralateral visual space. In the case of left-damaged patients, the 'hand effect' was observed, the deficit in reaching appeared for the right hand only, for targets in both visual fields. Additional symptoms in patients suffering from optic ataxia are the problems in avoiding obstacles and correcting the hand trajectory during reaching. In tasks where

the target position was changed during reaching, patients were not able to correct the hand direction on time and move along a smooth trajectory. Instead, the task was executed as a succession of two steps - moving toward the first target, then moving from the first to the second one. Although discussions which parietal regions are related to the emergence of optic ataxia still exist, the common believe is that this disorder appears as a consequence of lesions around the intraparietal sulcus and parieto-occipital junction, mainly corresponding to the superior parietal lobe. The reference [7] presents a detailed discussion of this matter and gives some additional arguments in favor of the relation between lesions on SPL and the optic ataxia.

Another typical disorder that appears in patients with parietal lesions is neglect, the inability to perceive sensations from the contralateral field and the incapability to action in the contralateral field. The patients lack the awareness within the personal and extrapersonal space contralateral to the lesion, sometimes they refuse to take food placed in the neglected space, sometimes refuse clothing and toilet of the contralateral part of the body. Patients, also, fail to represent the contralateral side of the space on drawings and avoid stimuli approaching from that side. The contralateral limb may be held in an unusual position, and if a paralysis of that limb exists, the patient will deny it. The neglect is related to the lesions on the inferior parietal lobe, and the pronounced contralateral nature of the disorder is well explained with a phenomenon observed for populations of IPL cells, the strong directional tuning toward the contralateral space, as reported in [7].

Apraxias are a class of deficits characterized by the inability to plan movements, from the incapability of following verbal commands, to the difficulties in performing sequences of movements. Some specific disorders from this class listed in the literature are ideomotor apraxia - the inability to execute a motor command and ideational apraxia - the inability to create a plan for a specific movement.

All these examples lead to the conclusion that PPC has an indispensable role in planning motor commands, for both eyes and hand, as a response to various sensory inputs. The detailed study of the properties of each of the regions identified within PPC is carried through electrophysiological recordings in monkeys performing reaching and delayed reaching tasks.

2.2.2 Regions of the posterior parietal cortex

The detailed anatomic division of the PPC and the functional roles of the identified regions are presented in this section. First, the commonly used nomenclature for PPC regions is described through an example of anatomical study. Alternative notations can also be found in the literature, but the detailed list of terminology is not relevant for this study. Instead, we will rely on the presented one and, when necessary, the additional terms will be introduced with the reference to the adopted nomenclature. This approach is sufficient for describing the main ideas related to the PPC, together with the motivation for this work.

Some relevant properties for each of the PPC regions are also described in this section: the cortical inputs and outputs, the signals that modulate cell activities, and the functional roles in the process of translating sensory inputs into motor outputs. The aim of this section is to illustrate a map of functional properties within the PPC, generally. The details about relevant computational properties and information coding mechanisms will be presented in the next section.

Architecture of the PPC

At the beginning, it is illustrative to present results of cortex regions identification based on the cytoarchitectonic differences. An extensive in-vitro analysis of monkey cortex slices has

been carried out by several researcher, however the overview of such studies is out of the scope of this work. Instead, the results from one recent example [52] will be presented in this section. We intend to visualize the cortex regions of interest, rather than to explain their architectonic specificities.

The study presented in [52] combines several architectonic tracers in order to precisely identify cortex regions and their subdivisions for macaque monkeys. Cytoarchitecture is one of the considered parameters here, but the study dominantly relies on the organization of fiber bundles in the layer 4 (or on the border between layers 3 and 5), the properties like density of myelination, or the appearance of pronounced radial or tangential components. Figure 2.2 summarizes the identified regions in the surrounding of the intraparietal sulcus, and on its banks and fundus (i.e. the bottom surface). The panel on the left shows the unfolded, 2D representation ('flat map') of the intraparietal sulcus (IPS) and nearby regions. The four surrounding sulci are also marked: parieto-occipital sulcus (POS), lunate sulcus (LuS), central sulcus (CeS), and cingulate sulcus (CgS) on the medial wall of the hemisphere. The figure shows 17 identified regions on the banks of the intraparietal sulcus and near the parieto-occipital junction. The four panels on the right correspond to the coronal sections marked by red lines on the left panel; they illustrate the vertical organization, from the most posterior part of IPS (section 1) to the most anterior one (section 4).

Before describing the topology of the considered region, it is necessary to introduce some common notation for directions. 'Medial' and 'lateral' refer to proximity to the medial plane, the separation plane between the two hemispheres; usually, when comparing two positions, medial one is the one closer to the medial plane, while lateral denotes the opposite one. Other standard notations from anatomy are also adopted here; dorsal, ventral, caudal and rostral, stand for 'direction toward the spine', 'toward the stomach', 'toward the tail', and 'toward the nose'. When describing a position inside the sulcus, dorsal and ventral mean 'toward the tip of the sulcus' and 'toward the fundus'.

Five regions are identified around the parieto-occipital junction. Two visual areas, V3 and V3A are separated due to the difference in density of myelination and the strong tangential fiber component present only in V3A. These two regions belong to the extrastriate cortex, a part of the visual cortex placed close to the primary visual cortex (also called the striate cortex), that corresponds to the Brodman areas 18 and 19. Additionally, the parieto-occipital (PO), posterior intraparietal (PIP) and lateral occipital parietal (LOP) regions are shown. The panel one on the Figure 2.2 shows the section through the most posterior part of IPS, nearby the junction with POS. The PO region occupies the medial bank of the sulcus, while PIP, V3 and V3A correspond to the medial, central and lateral part of the fundus, respectively. The lateral bank belongs to the LOP region.

Sections two and three examine the posterior and middle part of the IPS, and they are illustrated on the corresponding panels on the right. The second section passes through the region 7a, the subdivision of the Brodmann area 7, on the nearby gyrus lateral to IPS, and medial intraparietal region (MIP) on the medial bank of the sulcus. The medial dorsal parietal area (MDP) lays on the medial surface of the hemisphere, and its position on this figure is somewhat misleading. The third section passes through the central part of the IPS, showing the regions buried inside the sulcus, together with the Brodmann regions 5 and 7 on the surrounding gyrus. The dorsal part of the Brodmann area 5 (5D) is placed medially to the tip of the IPS, while area 7 lays laterally to the IPS. The regions inside the sulcus are depicted on the panel 3, the ventral part of the area 5 occupies the medial bank, the ventral intraparietal region (VIP), placed deeply inside the sulcus is divided into medial and lateral part, as well as the lateral intraparietal region (LIP) on the opposite bank of the sulcus. Finally, the fourth panel shows the organization of the most anterior part of the IPS, corresponding to the region 7b on the gyrus, the newly introduced region 7t on the tip of the sulcus, part of the region 5

on the opposite bank, and continuing to the anterior part of the PC, the Brodmann area 2 of the somatosensory cortex.

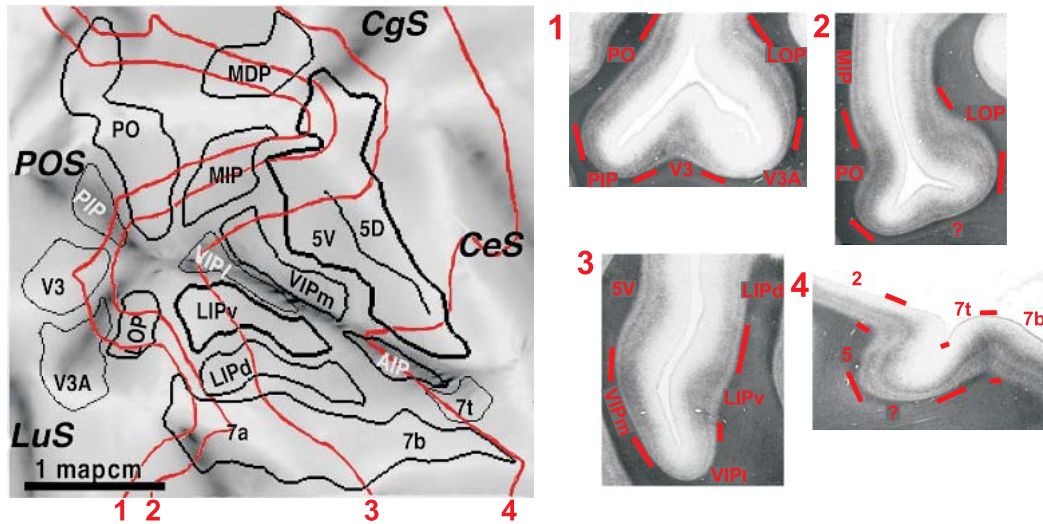


Figure 2.2: Parcellation of the intraparietal and the surrounding regions based on the architectonic traces. Left: The 2D representation of the considered cortex region, with the 'unfolded' intraparietal sulcus. The notation for the surrounding sulci: POS - parieto-occipital sulcus, CgS - cingulate sulcus, CeS - central sulcus, LuS - lunate sulcus. The identified regions are denoted as: PO - parietal-occipital area, PIP - posterior intraparietal area, V3, V3A - visual area 3 and 3A, LOP - lateral occipital parietal, MDP - medial dorsal parietal, 5V, 5D - Brodmann area 5, ventral and dorsal part, 7a, 7b, 7t - the identified regions within Brodmann area 7, MIP - medial intraparietal, VIPl, VIPm - ventral intraparietal area, lateral and medial part, LIPv, LIPd - lateral intraparietal area, ventral and dorsal part, AIP - anterior intraparietal area. The red lines mark the unfolded coronal sections of the tissue. Panels on the right show the original shape of these sections, together with the regions identified inside the sulcus. Figure is adapted from: J. W. Lewis and D. C. Van Essen, Mapping of Architectonic Subdivisions in the Macaque Monkey, With Emphasis on Parieto-Occipital Cortex, in *The Journal of Comparative Neurology*, vol. 428:79-111, 2000.

Figure 2.3 gives an alternative view of the PPC regions and their positions on the neocortex. The panel one shows the lateral view of the left hemisphere, together with the parietal regions 5 and 7 placed around the intra parietal sulcus (IPS). The subdivision of the area 7 into 7a and 7b is also marked. Additionally the primary motor region (M1) and the two parts of the dorsal aspect of the premotor region, the rostral (PMdr) and the caudal (PMdc) part, are illustrated. The second panel of the figure focuses on the regions buried inside the IPS, MIP and LIP on the medial and lateral bank of the sulcus, respectively, VIP occupying the medial and lateral parts of the fundus, and AIP in the anterior aspect of the IPS region. Finally, the third panel shows the medial wall of the hemisphere, together with the regions visible from this aspect - MDP, PO and 7m. The CiS denotes the cingulate sulcus position, while POS marks the parietooccipital sulcus. The presented illustration gives a more realistic, three-dimensional perspective of the parietal cortex organization. Still, the 'flat map' is more convenient for an illustration that aims to represent the maximum available information about the regions and their relative positions.

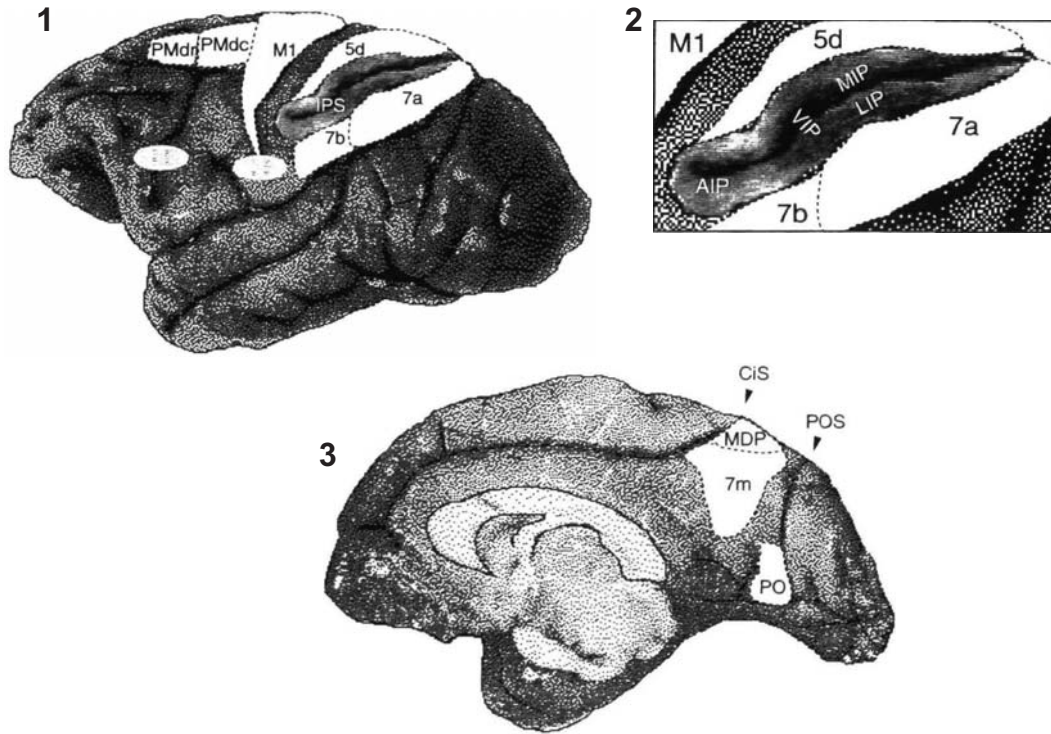


Figure 2.3: Illustration of the regions within PPC. Panel 1: the lateral view of the left hemisphere, the shaded area marks IPS, regions 5d, 7a and 7b of PPC are shown around the IPS, motor area M1 and the premotor regions PMdr (dorsal premotor rostral) and PMdc (dorsal premotor caudal) are also shown. Panel 2 presents the four regions buried inside the IPS: MIP, LIP, VIP and AIP areas. Panel 3 illustrates the medial view of the left hemisphere and the three regions placed on the medial wall: MDP, PO and 7m; the positions of the cingulate sulcus (CiS) and the parieto-occipital sulcus (POS) are also marked. The figure is adapted from R. Caminiti, S. Ferraina, P. B. Johnson: *The Sources of Visual Information to the Primate Frontal Lobe: A Novel Role for the Superior Parietal Lobule, Cerebral Cortex*, 1996.

Functional roles of the PPC regions

An overview of the main properties for each of the identified regions is given in this section. Most of the presented results are obtained by recording activity in the brain of monkeys, trained to follow visual (or auditory) commands. In a typical setup, a monkey is placed in front of a screen. Instructions controlling an experiment are presented on the screen. A chamber for recording electric activity is fixed to the animal's head during the whole experiment. Tips of one or several electrodes are inserted into the tissue, to record extracellular potentials. Finally, the spike trains are extracted from these signals and used to further analyze the typical cell behavior.

The passive response to a stimulus is recorded by simply applying the stimulus and monitoring the response. For analyzing the motor properties, the animal is trained to execute a movement following the instructions on the screen. The movement can be a saccade, i.e. a rapid eye movement, a hand movement, grasping for an object etc. Also, the movement can be restricted to 1D, 2D, or executed in 3D. Although, naturally, the reaction to the sensory input comes first, and an eyes movement precedes a hand movement, it is difficult to identify what influences each cell, when monitoring the reaching task. Therefore, many variations of

the same task have been presented in the literature. Introducing a delay between the sensory input and the movement separates the sensory and motor related signals. Multiple targets and movements are executed sometimes, in order to access the questions related to the memory and movement planning. Also, the animals were, sometimes, instructed to execute the eyes and hand movements in opposite directions, in order to disassociate the two types of movements.

The medial intraparietal region (MIP) is usually related to planning of reaching movements. Traditionally, it is considered the part of the area 5, and therefore, the part of the superior parietal lobe. Anatomical studies show that MIP receives inputs from regions within extrastriate or directly connected to extrastriate cortex, which supply it with visual information. The region mainly projects to the area near the border between the dorsal premotor (PMd) and primary motor cortex, and also to the rostral part of the PMd (PMdr); it also projects to the caudal part of the PMd (PMdc) [101, 15].

Visual, somatosensory or bimodal cells are reported in the MIP. The classification refers to the type of coded stimulus, where bimodal cells respond to both, the visual and the somatosensory inputs. Some of the visual cells respond only to the signals within the reaching distance ('near' signals). Also, it is noticed that the visual fields expand if the region that can be reached expands (for example when a monkey is holding a rake), thus being well adapted to the stimuli placed within a reaching distance. This region is believed to take part in spatial representation for control of arm movements, and providing signals for visually guided reaching for the motor-related regions [22].

The lateral intraparietal region (LIP) is a part of the inferior parietal lobe (IPL) in the monkey cortex. It receives inputs from the extrastriate cortex and is connected to the other visually related PPC regions. The LIP outputs are directed toward the centers responsible for executing saccadic movements (superior colliculus, frontal eye fields, pons) [87].

This region is often related to planning of eye movements toward both, the visual and auditory stimuli. On contrary, it is not very active during the planning of hand movements. The increased activity is observed in the presence of a salient visual stimuli, i.e. the input relevant for executing a movement, which probably represents the attention to the salient stimuli. The LIP cells code the entire visual field in the coordinate system centered in the retina. The auditory signals, together with the visual signals, are also coded in these coordinates, although they are naturally related to the head position. The computational mechanisms of LIP, such as the coordinate system used in LIP for representing stimuli, the remapping of the coordinates after a saccade, the coordinate transform, have been of interest in several experimental studies [21, 22].

The ventral intraparietal region (VIP) receives inputs from several visually and motion related areas, together with inputs from somatosensory regions, and projects to the area F4 of the premotor cortex, responsible for head and mouth movements. This region integrates visual and somatosensory information and initiates corresponding motor activities. VIP cells are often responsive to visual and tactile stimuli, with very similar receptive regions for both types of inputs. For example, cells that have receptive fields near the fovea also have tactile receptive field near the mouth, i.e. the most sensitive visual region is related to the most sensitive tactile region. One unusual class of VIP cells react only to stimuli within 5cm from the face. Some of the VIP cells are sensitive to vestibular signals (the signals indicating the position of the body in the space, the sense of equilibrium). The VIP activity can be interpreted as planning of actions in the perioral space (close to the mouth), thus helping in reaching with the mouth. Alternatively, this region may be responsible for supplying different

representations of space (head-related signals, vestibular signals etc.) to the premotor cortex, which enables calculating coordinates for a movement [22]. The VIP is also reported to play a role in self-motion through the environment, and estimating directions for self-motions [103].

The anterior intraparietal region (AIP) is believed to contribute to the hand shaping and grasping of 3D objects [74, 64, 22]. The populations of visually and motor responsive cells were found in this regions indicating that it contributes to control of visually guided movements. Unlike some other regions in the PPC, like MIP or area 5, the AIP is not involved in the representation of cue positions in the space, or visual guiding of a movement. Instead, the visually responsive cells in this region react to shape, orientation in space, and other 3D features of an object. This holds for both, the visual and the memorized objects. A study based on the delayed reaching task revealed the activity related to planning of the specific type of hand movements [64], indicating the role of the AIP in the control of hand shaping and grasping.

A part of the superior parietal lobe (SPL), *the Brodmann area 5* (also denoted as area PE in some studies) , is a motor-related region, playing role in the planning and coordination of hand reaching movements. It represents the source of most of the inputs to the motor and premotor regions of the frontal lobe, supplying the information necessary for initiation of the reaching movements. Still, until nineties, no evidence of the connections to visual areas was identified, i.e. the sources of visual signals in the area 5 were unknown (the 'parietal paradox'). The studies of the regions PO and 7m [15] revealed the indirect connections to the extrastriate cortex.

The first report of the cells in area 5 active during hand movements is presented in [62]. The cells were classified with respect to sensitivity to various sensory and motor signals. A large population of the examined cells reacted to passive rotation of the limbs, but they showed a prominent discharge only during the active hand movements toward the target. Also, a certain population of cells was unresponsive to any kind of passive stimuli, and started to discharge only during active hand movements. Later studies focused on analysis of mechanisms behind planning of hand movements, the spatial representation of hand and target positions, the appearance of intention and attention before a movement, etc. [12, 21].

The Brodmann area 7 is a part of the inferior parietal lobe (IPL) in the monkey cortex, and is usually anatomically divided into the regions 7a and 7b. The first of them is of particular interest for the presented work, and some of its properties are examined in the following sections. The area 7a receives multiple visual inputs, mainly from the areas V2, PO, the superior temporal sulcus and the dorsal prelunate gyrus. Area 7b receives somatosensory signals, and projects to the several prefrontal and frontal areas, including the PMv and supplementary motor cortex (SMA). The properties of the region 7a, presented in [9] and [10], will be reviewed in details in this chapter.

The area 7m of the medial parietal cortex is considered to be the early stage in the control of a visually guided reaching. It receives inputs from the visual areas of the occipitoparietal cortex, and projects to the dorsal premotor cortex [29, 15]. This region is also identified as a bridge between the superior and inferior parietal regions, which explains the 'paradox' of the parietal lobe. In other words, it connects the area 7, receiving inputs from the visual regions but having no direct connections to the motor regions in the frontal cortex, with the area 5, the motor-related area directly connected to the frontal motor regions, but receiving no direct inputs from the visual cortex [15]. The various types of cells with combinatorial properties were identified in 7m. Many of them are influenced by the eye-position signals, but the influence of the hand position, and hand movement signals was also observed. This

supports the hypothesis that 7m contributes to the planning of motor commands for hand movements, or visual monitoring of hand trajectories [29].

The medial dorsal parietal region (MDP) is also a part of the cortical network for the control of visually guided movements. This region, together with the area 7m, receives the visual inputs from the parts of the extrastriate cortex, or the regions directly connected to it. The outputs are projected to the dorsal premotor cortex of the frontal lobe. These anatomic observations indicate the role of this area [101].

The parieto occipital region (PO), sometimes also denoted as V6A, is placed on the medial side, ventrally to the MDP and 7m. It receives direct inputs from several pre-striate regions. Therefore it is strongly influenced by visual signals, and is considered as one of the visual areas in the PPC. The PO projects to the PMd directly, and indirectly through 7m and MIP. A characteristic of the cells in this region is that they have large visual receptive fields, related to coding of peripheral, rather than foveal signals. They are also orientation and direction selective, and probably contribute to the ambient vision and target detection. The activity of most of the cells is modulated by the eye position, although a minority of them are independent of the gaze angle [101].

In the studies of Andersen and others [4, 21] *the parietal reach region (PRR)* is analyzed in the context of the hand reaching movements. This area corresponds to the regions MIP, 7a and the dorsal aspect of PO. Its computational mechanisms for coding of visual and auditory targets, for transformation of coordinate systems and for planning of hand movements are discussed in details in [21].

2.2.3 Electrophysiological studies

The studies on patients with parietal lesions indicate the role of the PPC as an interface between sensory processing and motor and premotor regions. The PPC receives sensory inputs of different modalities, visual and auditory signals, somatosensory inputs and information about the positions of limbs, hands and eyes, and plays a crucial role in early stage of planning motor activities, generating commands for movements of hands, limbs and eyes on a more abstract level [62]. The mechanisms behind these activities have been the object of interest for many studies [62, 32, 4, 22, 101], but throughout this work, we will concentrate on the identification of intention, the early motor plan, in the PPC. Some other relevant issues, the directional tuning in cells, the problem of coordinate transform and signal representation, the specific functions of certain parietal regions, and the discussion about the computational framework operating in the cortex, will be briefly reviewed in this section.

Preferred direction and population coding

Relation between a single cell discharge and the spatial position of the visual stimuli is observed in several neocortex regions, as well as in some other structures within the human brain. Motor cortex [33, 34], premotor cortex [99, 32], parietal regions 5, 7 [43, 32, 9] are some of the neocortex regions where the direction-sensitive cells were reported.

Typically, in the experiments with visual stimuli presented in 2D, a cell activity corresponds to a bell-shaped curve, the function of the target position described as an angular distance from the referent direction. The activity is maximal for the target position corresponding to the preferred direction, and decreases as it moves away from the preferred

direction. This dependency can be described by a cosine function, as in [34]:

$$r(\theta) = A + K \cdot \cos(\theta - C) \quad (2.1)$$

where $r(\theta)$ is the frequency of cell discharge, and a function of the angular position of the visual cue, θ , in the 2D coordinate system. The parameters A , K , C can be determined by fitting this expression to the data. The parameter C represents the preferred direction, in the same coordinate system used for expressing θ . A slightly more complicated model is presented in [9], and given by the expression:

$$r(\theta) = \begin{cases} A + K \cdot \cos(x \cdot S) & \text{if } |x \cdot S| < \pi \\ A - K & \text{elsewhere} \end{cases} \quad (2.2)$$

where $x = \arccos(\cos(\theta - C))$. The additional parameter S controls the width of the tuning curve, for $S = 1$ the expression reduces to (2.1), $S < 1$ corresponds to a broader tuning function, while $S > 1$ results in a sharper one.

The majority of cells possess broad tuning curves, which indicates that a single cell activity cannot reliably represent a visual cue position. The mechanism of coding directions in the brain rely on populations of cells. Indeed, a weighted activity over a population gives a good estimation of the target direction. A randomly selected set of cells with preferred directions spanning the 2D space can be used for a very accurate decoding of target positions. This has been demonstrated for the motor cortex [34, 32], given the expression:

$$\tilde{\theta} = \sum_{i=1}^{N_{cells}} C_i \cdot (r_i(\theta) - b_i) \quad (2.3)$$

where N_{cells} is the number of cells in the population, and C_i and $r_i(\theta)$ represent the preferred direction and frequency of spiking of the i -th cell, respectively. The parameter b_i is a constant, a parameter of the model. Finally, θ indicates the visual cue direction, and $\tilde{\theta}$ the estimated direction from the population activity.

This result provides a basis for the brain-machine interface applications, using the activity recorded in the motor cortex [20, 83, 92, 65, 11], in the dorsal premotor cortex [77], in the parietal cortex [72, 84], or in more than one site chosen on the frontal and parietal lobe [16, 17].

Spatial information representation and coordinate transformations

The role of the PPC as the interface between sensory and motor related areas, receiving sensory inputs of different modalities and generating motor outputs, implies that some sort of mechanism for coordinate transform has to exist in this region. Cells from some of the PPC regions respond to visual, auditory, somatosensory signals, and can be modulated by the eyes position, neck proprioceptive signal, i.e. the position of the head with respect to the body, or vestibular signal, i.e. the position of the head with respect to the world coordinates [5]. For example, many cells in the area 7 respond to visual stimuli, both static and moving; cells in the area 5 are less responsive to this type of signals but often react to passive rotation of joints or active reaching by hand [62]. Areas 7b and VIP are more related to the somatosensory signals [5, 22], while the LIP activity depends on both, the visual and the auditory signals [5, 21]. Regions like MIP and MDP are strongly influenced by hand movements. Just this short list of various signals present in the PPC implies the need to use several reference frames - visual signals are often related to the retinotopic coordinate system, and influenced by the eyes position in that coordinate system. Auditory signals are related to the head-centered coordinates, while planning of the hand movements has to include signals coded in the hand- and body- centered coordinates. The experimental studies carried in this field

revealed roles of different PPC regions, still there is no consensus about the mechanism for coordinate transformations on the entire path from sensory inputs to motor outputs.

At the beginning, the two concepts necessary for describing the coordinate transforms mechanisms in the PPC, namely the notions of reference frames and gain field modulation, are introduced.

The reference frame denotes a set of axes used to represent the location of an object, in terms of distances from these axes. The identified reference frames in the PPC are the retinotopic, head-centered, limb-centered, body-centered and world centered ones. The retinotopic coordinate system is centered in the point of visual fixation, i.e. an object position is represented with respect to the retina position. Naturally, with each saccade, the coordinate system is shifted and the mechanism for updating coordinates after each saccade exists in the brain. Coding the object position with respect to the head position is also observed in the parietal cortex, for the fixed head the coordinate system does not change, but after a head movement the coordinates has to be updated to compensate for the shift. The coordinate system related to the hand position is necessary for the planning of hand reaching movements. Here, all the objects are coded with respect to the initial hand fixation point. The same way, in the body-centered coordinate frame the object position is described relative to some fixed point on the body. Finally, the world-centered coordinate system refers to the surrounding space.

The gain field modulation is the key mechanism in the parietal cortex that enables coordinate transform from one reference frame to another. This is a multiplicative effect that modulates a cell discharge according to certain relevant signals, e.g. eyes position, hand position etc. The gain field is observed in many experimental studies [21, 4], and the mechanisms behind it are also studied through computational models [75].

The Figure 2.4 illustrates coding in different reference frames and the effect of gain field modulation. The cell response field, i.e. the frequency of the cell activity as a function of the visual cue direction, is represent as a bell-shaped curve that has its maximum in the preferred direction of the considered cell. The target position is expressed as the angular distance from a certain reference direction (zero degrees on the figure). The x-axis corresponds to the head-centered coordinate system, used to represent the target position (in degrees), while the y axis represents the firing frequency (in Hz). The panel **a** illustrates the eyes-centered coordinate frame. The red, blue and green lines correspond to the visual response field when the eyes fixate points at -10, 0, and 10 degrees. Since the reference frame is eyes-centered the entire response field is translated together with the fixation point. On the next panel, **b**, the limb-centered reference frame is illustrated, and the three lines correspond to the receptive field for different hand positions. Here, the point of hand fixation represents the center of the coordinate system, therefore, when the hand moves, the entire receptive field is translated to preserve the hand-centered coordinates. The panel **c** shows the head-centered reference frame, and the three receptive fields correspond to the three different head fixation points. Since the x axis shows, exactly, the head-centered system, for every head move, the entire panel **c** 'moves', and the receptive field preserves always the same coordinates. Finally, the panel **d** illustrates the intermediate coordinate frame, with the three lines again corresponding to the three eyes fixation points. Here, the response field is translated for each eye movement, but with a smaller shift than the eyes. Therefore, this reference frame reflects the influence of some other signals, together with the eyes position. The last panel presents the case of eye-centered coordinate frame with the eye-position gain. The same translation of the response field in accordance with the eyes translation, as on the panel **a**, exists here. The difference is that, not only the position of its center, but its amplitude as well depends on the eyes position. In this example for a smaller angular position of the eyes the activity is less intensive.

The last panel of this figure demonstrates the mechanism of coordinates transform from the

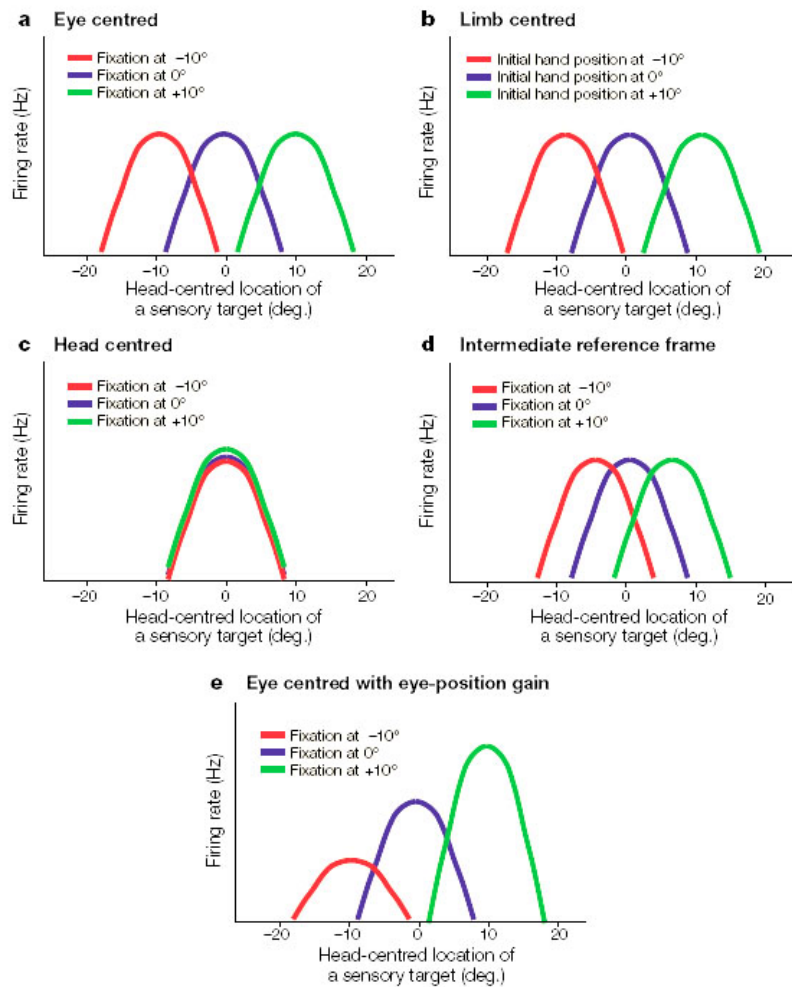


Figure 2.4: The illustration for reference frames and gain modulated signals. The figure reproduced from Y. E. Cohen and R. A. Andersen: A Common Reference Frame for Movement Plans in the Posterior Parietal Cortex, in *Nature Reviews Neuroscience*, 2002, vol. 3, pp. 553-562 (see [21]). A cell response field is represented by a bell-shaped curve which maximum corresponds to the preferred direction for that cell. The x axis represents the target coordinates in the head-centered reference frame expressed as the angular distance from some referent direction (in degrees). The y axis shows the firing rate of a cell activity (in Hz). The panels **a**, **b**, **c**, **d** explain the eye, limb, head and intermediate reference frame, respectively. The last panel **e** illustrates the effects of gain modulation.

eyes-centered to the head-centered system. The initial coordinates refer to the eyes position, while the amplitude of the cell discharge depends on the position of the eyes in the head-centered reference frame. This provides sufficient information for the read-out cells population to recalculate a new set of coordinates, related to the head position. In general, the gain field is the mechanism to represent the information about more than one reference frame in the same population of cells. The cells that receive these inputs are, then, able to recalculate the coordinates in the new coordinate system. The translation of coordinate frames and the gain field modulation are also studied on the computation models, presented in [75, 102].

The proposed models of the hand-eyes coordination in the PPC

Although the questions of reference frames and gain field have been studied intensively, there is still no consensus about the unified mechanism that translates the visual and auditory signals from their original coordinates to the hand-centered system necessary for hand navigation. The literature [4] refers to the three methods proposed by different research groups. The first of them proposes the sequential transformation of coordinate frames on the path from visual inputs to the motor output signals [30, 60]. The visual inputs, coded in the retinotopic reference frame and modulated by the eyes position with respect to the head, are first transformed into the head-centered system. This one is, then, further transformed into the body-centered coordinates using the proprioceptive signal, i.e. the position of the head with respect to the body. Finally, knowing the hand position in the body-centered system, the target coordinates can be translated to the hand-centered reference frame. The evident problem with this method is its computational inefficiency, since it requires many stages and separate computations.

The method described in [12, 4], proposes a direct transformation from the eyes-centered to the hand-centered coordinates. It assumes that the parietal cortex receives the information about the target and the hand positions in the same, retinotopic, coordinate system. Then, the direct subtraction of the hand coordinates from the target coordinates translates the target in the hand-centered reference frame. The experimental support for this hypothesis is found in the recordings from area 5, where the eye-centered and hand-centered reference frames were observed simultaneously. Also, the eye-centered coordinates in the PRR are modulated by hand position represented in the eye-centered reference frame.

The combinatorial method, discussed in [8] and [7] assumes no modular organization in the PPC computations. The retinal target location, eyes, hand, and head positions are all combined together and the limb centered coordinates are read at the output. The problem of this method may be the 'curse of dimensionality', i.e. if the same set of cells have to represent various stimuli at the same time, the number of possible combinations of signals increases exponentially as a function of the number of stimuli to be represented. On the other hand, the new experimental evidences, given in [9, 7], show that the majority of cells in the SPL have a global tuning field (GTF). In several different reach-related tasks, and during various epochs of these tasks (some of them related to the eyes movement, other related to the hand movement), the preferred directions were calculated for the recorded cells. For the majority of cells the different preferred directions are clustered within a small region in the space, the phenomenon named the global tuning field. This indicates that the preferred direction of a cell does not vary much with the change of eyes or hand position, which indicates the allocentric reference frame rather than the eyes-centered one. The allocentric system is not related to either eyes or hand but to some referent point independent on both of them. This indicates that the combinatorial method may explain the experimental evidence better than the previously proposed ones.

Intention coding

In the neurophysiological literature, the (motor) intention is defined as an early plan for movement that specifies the goal and the type of the movement but does not, necessarily, contain the explicit commands for the motor neurons, necessary for executing that movement. It is a high level and abstract representation of a movement [4].

The presence of intention is more evident in the delayed reaching, than in the classical reaching tasks. During the delay period, after the processing of the visual information is completed, and before the movement starts, the persistent activation of the cells is observed.

It is shown, in various experiments, that this activity does not reflect neither the sensory memory nor attention, but the intention for a movement.

The experiments with two targets, instead of only one, indicate that it does not correspond to the sensory memory [4]. Here, the monkey is instructed to memorize two targets, and reach for them sequentially. The persistent activity is present only for the next target to reach, although both of them have to be memorized at that time. Therefore, the activity of the delay period does not reflect the sensory memory.

Distinguishing between the intention and the spatial attention in the experiments is a more difficult task. In order to assess this question, the experiments that require two types of movements have been introduced. The target position is presented to the monkey at the beginning of the experiment, and during the delay period it is memorized, and the movement is planned. After the delay, the animal is instructed to perform the movement. The only difference between the two modes of the task is the required movement, in one case it is a saccadic eye movement, and in the other the hand movement. The hypothesis is that in both cases the spatial attention and the sensory memory have to be the same, the only difference is the type of movement, and consequently the related planning or intention. The results obtained by recording the cell activity in LIP and PRR indicate that the delay period activity indeed reflects the intention. Typically, the cells recorded in the LIP were much more active during the saccades, while the PRR cells responded more to the hand movements, for the same target position. As described before, the LIP cells are believed to take part in the planning of saccades, while the PRR region is related to the hand movements. The outcome of the experiment supports the hypothesis of the presence of intention in the parietal cortex [4].

2.3 Behavioral Experiments

In this section, the framework and motivation for our study are presented in detail. An extensive set of behavioral experiments is designed to collect a database analyzed throughout this work. The measurement apparatus, the details regarding the recording site and the organization of the collected database are described at the beginning of the section. The results of the first analysis conducted on this database, i.e. the study preceding our work, are also presented here. A more detailed discussion of these results can be found in the literature [9, 10]. This work, together with the additional observations regarding population coding of intention, computational properties of the parietal cortex and brain-machine interfaces was the motivation for our study; the arguments that will be presented at the end of this section.

2.3.1 Measurement apparatus

The extracellular recordings of the electrical activity in the cortex were collected using a multielectrode recording system, consisting of seven microwires, described in details in [26, 6]. The second reference describes the complete scheme for automatic control of electrode positions, collecting and storing the recorded signals, on-line identification of cells and spike trains; this illustration is reproduced on Figure 2.5.

The lower right corner of the figure illustrates the recording system, and the animal position. During a session, the monkey is fixed to the chair in order to minimize movements, and to provide the support for the positioning system for electrodes (Eckhorn microdrive) connected to the recording chamber. The chamber holds the electrodes, and protects the tissue between the recording sessions. The particular system, used here, contains seven electrodes arranged linearly, i.e. seven recording channels. The collected waveforms are amplified, filtered and stored on the medium in the center of the scheme, for the off-line analysis. The

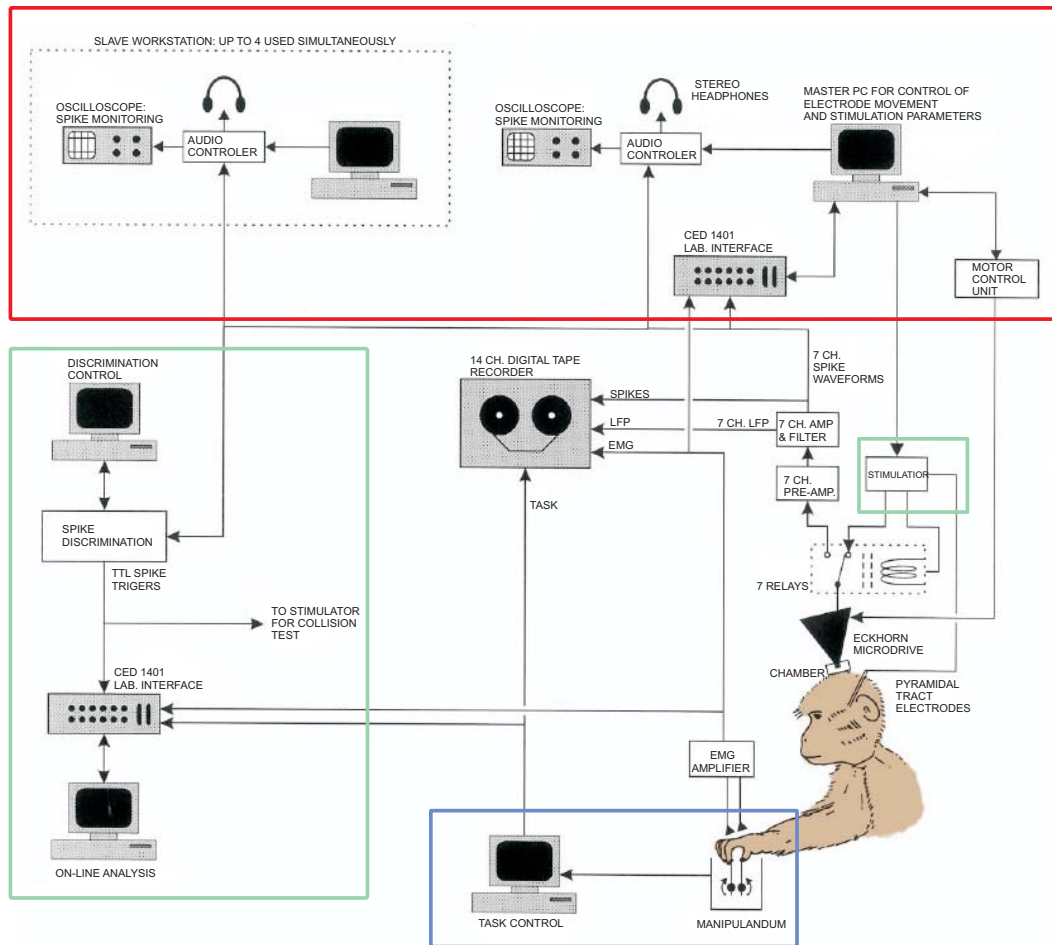


Figure 2.5: The schematic representation of the measurement apparatus. The lab monkey with the implanted chamber and electrodes, and the system for collecting recordings and stimulation of brain regions (if needed) are shown in the lower right corner. The apparatus for controlling and monitoring experiments is shown in the red box. Particularly important is fine tuning the electrodes position. The part of the system for on-line cell identification is marked with the green box. The data can be analyzed off-line, after being stored on the medium in the middle of the scheme. The blue box on the bottom shows the system for presenting task instructions to the monkey.

The figure is adapted from S. N. Backer, N. Philbin, R. Spinks, E. M. Pinches, D. M. Wolpert, D. G. MacManus, Q. Pauluis, and R. N. Lemon: Multiple single unit recording in the cortex of monkeys using independently movable microelectrodes, *Journal of Neuroscience Methods*, 1999, vol. 94, pp. 5-17.

local field potentials can be measured, as well as the raw extracellular electrical activity, the latter is sent to the system for identification of spike waveforms and spike times.

The green box on the left encircles the part of the system for the on-line spikes discrimination. A simple and fast double amplitude-time window algorithm is implemented in real time. This approach requires manual selection of one or several windows in the time-amplitude space, which are used to analyze the shape of spike waveforms. All the spike waves passing through all these windows are assigned for the same cell. Clearly, such a set of windows have to be determined for each of the channels, making it difficult to monitor many channels simultaneously. The described system permits identification of up to two cells from one chan-

nel. On-line algorithms are typically less reliable than the off-line ones, but they are useful for control of the electrode positions, maximizing the chance for successful recordings. The parameters for the identified cells (i.e. window sizes in 2D) are sent to the controller and to the system for stimulation.

The stimulation system is marked with a small green box on the scheme. It enables injecting the electrical pulses into a certain place in the brain and monitoring the responses on the recording site.

The controls for the entire system are shown in the red box. On the right is the central unit, including a PC controlling the electrode movements and parameters for stimulator. It is connected to the motor control unit for fine tuning of the electrode positions in 3D, and to the oscilloscope and audio controller for monitoring the recorded signals. Since one person can hardly maintain the entire process, up to four identical “slave” systems are added (on the left).

Finally, the blue box on the bottom shows the system for tasks control, the screen for presenting instructions and the manipulandum for reaching the target. The muscle activity, electromyography (EMG), can also be recorded and stored.

The implanted electrodes are glass-coated tungsten-platinum fibers with 1-2 $M\Omega$ impedance at 1kHz, the seven microwires are arranged in one row. The set of recorded signals contains the spike trains from up to fourteen cells and the recordings of hand and eye positions during the tasks. The hand position is monitored with a 21 inches touch-sensitive screen, also used to display instruction signals, while the eye position tracking required implanting an eye-coil. The head of the monkey was fixed to the chair holding the eyes 17cm far from the screen during all the experiments.

2.3.2 The position of the electrodes

In a series of behavioral experiments the neural activity of the 7a region of the inferior parietal lobe (IPL) was monitored in the left hemispheres of two monkeys. The figures 2.2 and 2.3 illustrate the position of the 7a region on the “unfolded” map of the IPL, and on the standard representation of the cortex, respectively. The precise position of the recording site, together with the positions of penetrations across all the sessions, is shown on 2.6.

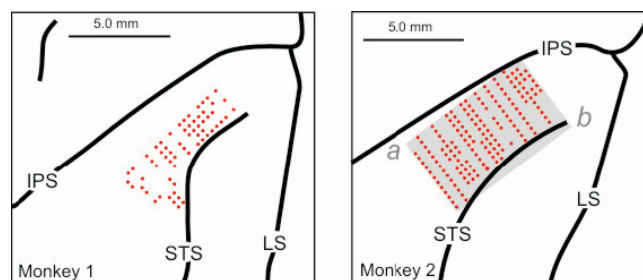


Figure 2.6: The precise position of the recording site. The three surrounding sulci are shown - the intraparietal sulcus (IPS), the superior temporal sulcus (STS) and the lunate sulcus (LS). The red dots on the figure show the exact positions of the electrode penetrations.

Adapted from: A. Battaglia-Mayer, M. Mascaro, and R. Caminiti: Temporal Evolution and Strength of Neural Activity in Parietal Cortex during Eye and Hand Movements, *Cerebral Cortex*, June 2007, vol. 17, pp. 1350-1363.

The three sulci defining the region are marked on the figure, the intraparietal sulcus (IPS) that separates the superior and inferior parietal regions, the superior temporal sulcus (STS) between the temporal and parietal lobe, and the lunate sulcus (LS) between the parietal and

occipital lobes. The positions of electrode penetrations are marked with red dots. In total, 559 neurons were recorded from both monkeys. The penetrations were confined to the exposed part of the IPL, perpendicular to the cortex surface, with average depth of 2mm.

2.3.3 The experiments

The set of behavioral tasks was designed in order to investigate the eye-hand coordination in the parietal cortex. Two rhesus monkeys were seated in front of a touch-sensitive screen, and trained to follow the instruction signals presented on the screen. The experiments and the methodological approach are described in details in [9] and [10].

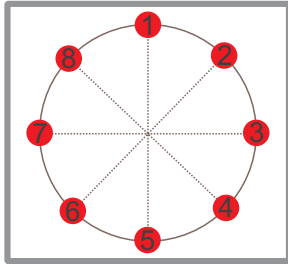


Figure 2.7: The eight target positions, as defined by the experimental protocol. One of them is illuminated during an experiment to signalize the goal for a reaching movement.

At the beginning of a typical task trial, a red light is turned on in the center of a touch-sensitive screen. The animal is required to fixate and touch this target for a variable period, indicated as CT (Control Time). At the end of the CT, the red light goes off, and the monkey is required to follow a given, task specific, instruction. A visual target signal appears in one of eight peripheral positions. The positions allowed by the experimental protocol are placed on a circle centered in the center of the screen, at 45° intervals from each other. The Figure 2.7 illustrates these positions, as they would appear on the screen. Typically, each task is executed for variable target positions, and several repetitions. One complete block of data consists of four trial replications, for each of the 8 target positions, in each of the first six behavioral tasks, plus three repetitions for sixteen target positions in the last task shown in 2.8.

This figure shows the schematic representation of the tasks, with the list of epochs corresponding to each of them. The tasks are denoted as Reach (R), Reach-Fixation (RF), Memory Reach (MR), Memory Reach-Fixation (MRF), Memory Eye (ME), No-Go (NGO) and Visual Fixation (VFIX) and will be described in details in what follows.

Reach task: As soon as the target appears, the monkey reacts naturally, first moving the eyes toward the target, then the hand follows in a coordinated way. The first row on the Figure 2.8 depicts this task. Each box corresponds to one epoch, a time-interval related to execution of one activity necessary for completing the required movement. During the first epoch, CT, as described before, the animal fixates the center of the screen with both, the eyes and the hand waiting for the further instructions. Immediately after the central light goes off, the target is illuminated and the preparation for the movement starts. The RTeh (Reaction Time for eyes and hand) reflects this preparation. Next, the movement of the eyes, with the hand still fixating the center, is executed during the MTe (Movement Time for eyes). Although the hand is still immobile here, the preparation for its movement is carried on. Therefore, signals related to the eye movement and the preparation for hand movement are present in this epoch. The additional preparation period for the hand is marked as RTh (Reaction Time for hand), and the movement happens during the MTh (Movement Time for

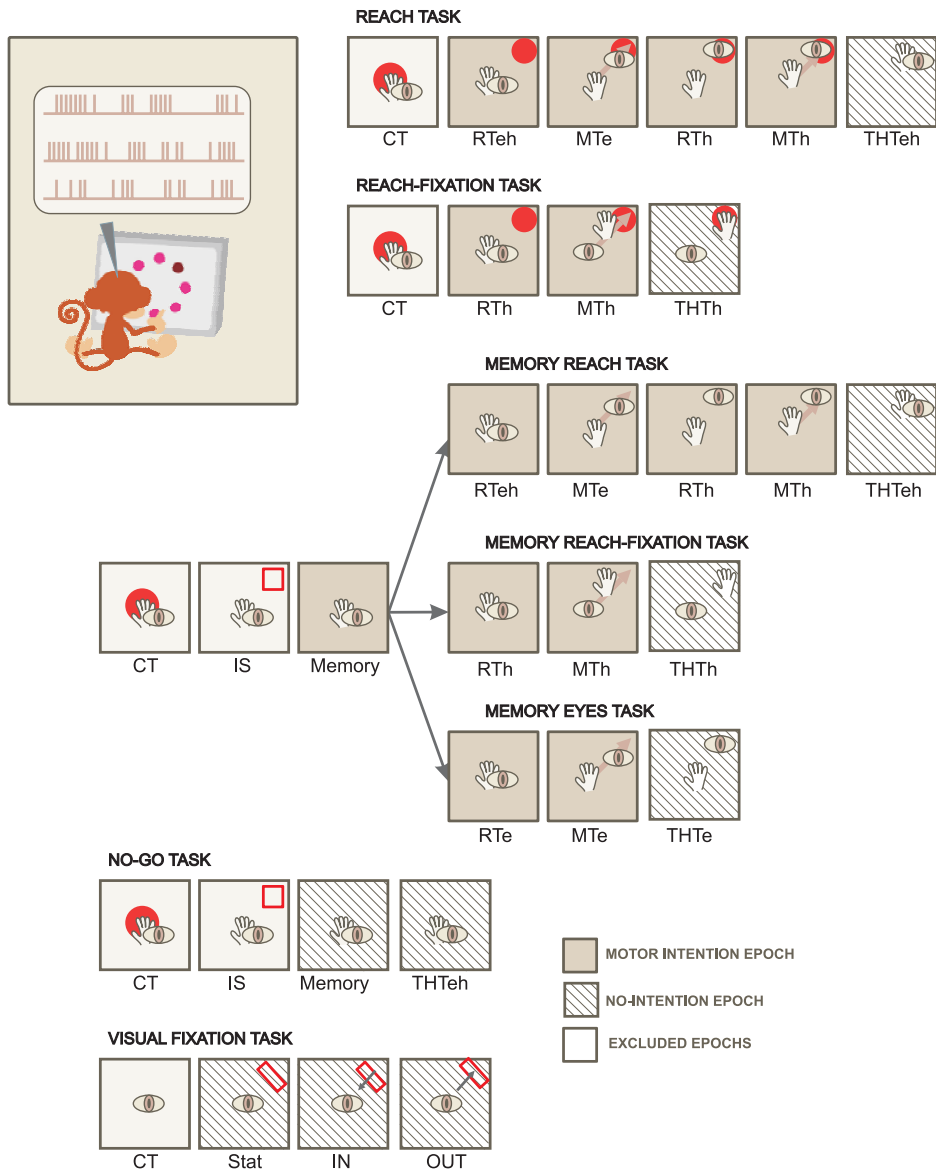


Figure 2.8: The schematic illustration of the tasks protocol. Each row corresponds to one task, and each box represents one experimental epoch. The instruction signals, eyes and hand positions are shown for each of the epochs. During the shaded epochs, the cells activity, at least for some of them, reflects motor intention.

hand). Finally, during the THTeh (Target Holding Time for eyes and hand), the animal is required to fixate the target for a certain time. Small letters in the names of the epochs are used to highlight the relation to the eye or hand activity.

Reach-Fixation task: The protocol for this task requires a fixed position of the eyes at the center of the screen, from the beginning till the end of the trial. After its presentation, the monkey locates the target position by the peripheral vision, and moves the hand only toward it. Therefore, a coordination between the eyes movement and the hand movement is not required by this task, but rather a decoupling of eye and hand motor behavior. The task, as shown on the Figure 2.8 consists of CT, RTh (reaction time for the hand), MTh (movement

time for the hand) and THTh epochs.

Memory tasks: The three Memory tasks were designed to monitor the cell activity during a reaching movement in absence of a visual target, i.e. for the target position saved in the working memory. Each of the three tasks start with the presentation of an instruction-signal (IS) in one of the eight positions. After 300 milliseconds, this signal goes off, and the monkey must remain immobile for a variable memory-delay time, until the go-signal appears on the screen. This period, denoted as 'Memory' on the Figure 2.8, corresponds to the time necessary for completing target memorization processes in the brain. The persistent activity observed during this epoch is often related to the first step in movement planning and the emergence of motor intention [4]. After the go-signal, the monkey is required to move eyes or hand toward the memorized target. Memory reach (MR) and Memory Reach-Fixation (MRF) correspond to R and RF tasks, respectively, with the target position saved in the working memory instead of being present on the screen for the entire task duration. Memory eye (ME) task is somewhat similar to MRF, since it also decouples hand and eye related activities. Here, the eyes move to the target while the hand remains immobile in the center. The selection of the specific memory task is dictated by the color of the instruction-signal; blue for the MR task, green for MRF and yellow for ME. At the end, the monkey is required to stay at the final position during the THT period.

No-Go: In this task, the eyes and the hand are immobile during the entire trial. The neural activity is recorded in a behavioral condition where the instruction signal does not call for any future movement, but requires both eye and hand to remain on the center position throughout the entire experiment. The target signal is presented during the IS interval, and its purple color indicates the NGO task. The last epoch, THTe, starts with the second instruction signaling the end of the memory time. In the three Memory tasks this serves as the go signal, but here it is an indication to remain immobile. Therefore, movement planning or motor intention should not be present in the recorded activity, while spatial attention and visual cue memory are expected, as in the other tasks.

Visual-fixation: This task is somewhat different from the previous ones since it serves for monitoring cells activity as a response to the visual cue presentation. Therefore no hand movement and, consequently, no hand-eye coordination is anticipated in this task. A visual stimulus is briefly flashed on the periphery of the visual field, and then moved at constant speed inward toward the fovea and outward from the fovea to the periphery while the monkey fixates the center of the screen. It can appear in one of sixteen positions (at $22,5^\circ$ angular intervals), instead of eight as in the other experiments. The first epoch, CT, is the period of waiting for the target signal to appear on the screen, as usual. The epochs 'Stat', IN and OUT correspond to the static presentation of the visual cue, the movement from the periphery to the center, and the return from the center to the initial position, respectively. In the framework of this study, this task is relevant since it should not involve planning of a movement and the emergence of motor intention.

Motor intention in the experimental epochs: In the Section 2.2.3 we introduced the motor intention in the context of neurophysiological studies, and presented the experimental evidences that it influences the neural activity in the parietal cortex. Starting from that definition, and knowing the key processes related to each of the described experimental epochs, we can identify those among them where the neural activity is likely to reflect motor intention. The boxes that represent epochs on the Figure 2.8 are colored in gray with respect to our expectation of absence and presence of motor intention. During the epochs involved in planning or executing a movement the activity of some cells might encode this information.

They are colored on the figure and will be relevant for the analysis presented in the next chapters. The epochs where the presence of motor intention is not expected are hatched, and those where it cannot be reliably identified in the cells activity remain white. As shown, the R, RF and Memory tasks probably encode the motor intention during some epochs, while NGO and VFIX should not contain it at all. In the first epoch, CT, it is difficult to identify specific signals in the brain. No go signals that can trigger motor intention precede this epoch, and a hidden planning of a movement cannot be reliably identified. The IS epoch is dominated by strong cell responses to presentation of a visual cue, the activity that can mask all other processes present in this time step. It remains white for being inconclusive. Finally, in the THT epochs the task is completed, and the recorded activity should not reflect any motor intention. Additionally, the end of each epoch is removed to compensate for a potential planning of a next movement. Still, a possibility for covert plans in this epoch cannot be ruled out completely.

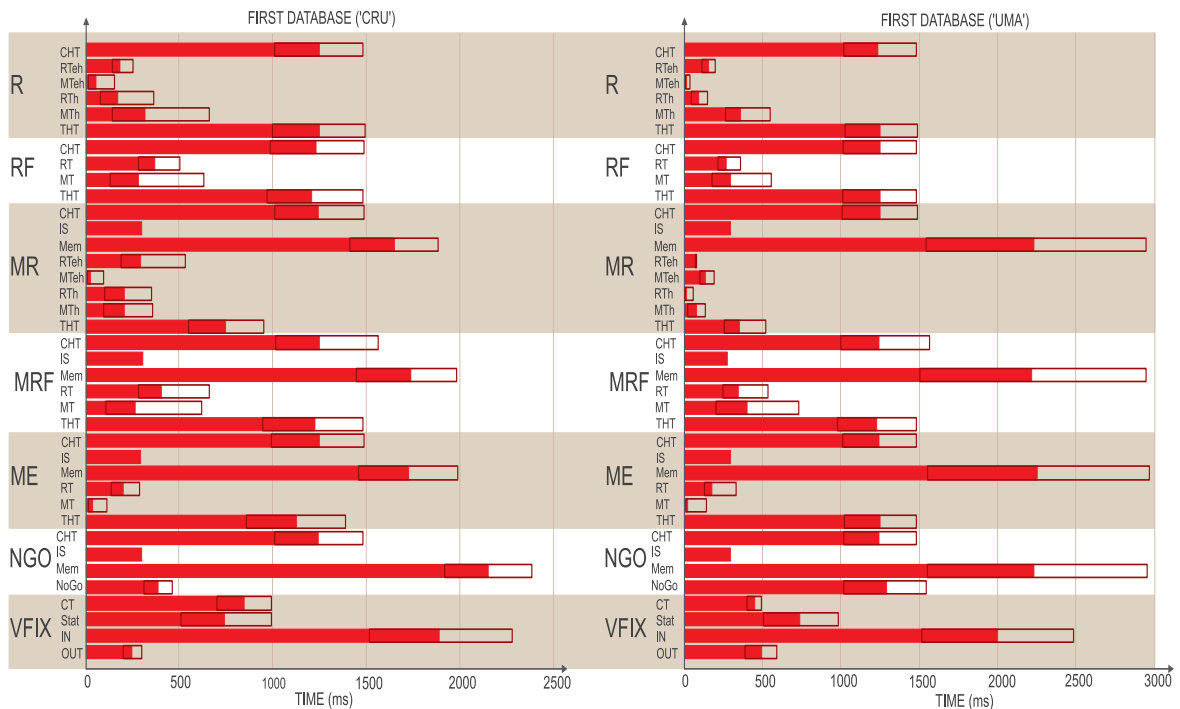


Figure 2.9: The average epoch length: left panel corresponds to the database for the monkey 'Cru', right for the monkey 'Uma'. All the epochs for the seven tasks are listed on the y axis. The x axis shows time in ms. A red bar shows the average duration of an epoch, calculated for different recording sessions and experimental conditions (target, trial). A dark red box marks the range of values that the corresponding epoch takes in all the examples in the database.

The average length for the epochs is marked on the Figure 2.9. Each horizontal red bar corresponds to the mean epoch duration, calculated across all the sessions and experimental conditions. Each dark red box shows the range of values that the corresponding epoch takes in various sessions. The x axis represents time, while the y axis lists the epochs. The control time at the beginning of each experiment is an interval of variable duration of 1-1.5s. Reaction and movement times last less than 1s. The instruction signal is presented for 300ms at the end of CT in the Memory and No-Go tasks. In the VFIX task the static target is presented for 1-1.5s, then it is moved first inward, then outward during the epochs marked as IN and OUT. The Memory delay period lasts for 1-3s. At the end of the experiment the animal is

required to fixate the target for a variable interval of 1-1.5s (the THT epoch).

2.4 Database description

Two databases of recordings, each collected from the left hemisphere of one monkey, are available as outcome of the described experiments. The schematic representation of one of them (from the monkey 'Cru') is given in Figure 2.10. The presented example consists of forty five 'files', while the second database (for the monkey 'Uma') contains fifty seven of them. The term 'file' denotes all the recordings collected from the same site, i.e. for a fixed electrodes position and the corresponding set of cells. All the cells are monitored under all experimental conditions, defined by the protocol. In other words, one 'file' contains recordings for all seven tasks and all allowed target positions for those tasks; R, RF, Memory tasks and NGO are repeated for eight target positions, and VFIX for sixteen on them. Four replications of trials for each combination (Task, Target) are completed for the first six tasks, while VFIX is repeated three times per target position. The three middle columns on the Figure 2.10 illustrate the tasks, target positions and trials for each file, while the last panel shows all the data collected in one trial, for one target position and task, and we will refer to it as to one data set.

Each data set within a file contains at most fourteen recorded spike trains, since from each electrode recording, neural activity from up to two different cells is collected. The corresponding epoch marks, i.e. markers denoting the beginning and the end of each behavioral epoch within a trial are also saved. Some markers were determined by recording eyes and hand position in each trial. The recordings of eyes and hand position are also collected, the hand position using a touch-sensitive screen and the eyes position from the implanted coils. The continuous 2D trajectories are discretized with 10ms time step.

An extract of the recordings from the database is given on the Figure 2.11. The example shows the activity of the first cell from the file obtained in the ninth recording session on the monkey 'Cru'. A spike train is depicted as a sequence of short black bars, marking spike occurrences. The shorter red bars at the top of each spike train show the beginning and the end of epochs for the considered task. A panel shows all the recordings obtained for one task. Target position is marked on the y axis, next to the corresponding spike trains obtained in four repetitions (three for the VFIX task). All spike trains are aligned to the control time (marked as CT of the figure), and the x axis shows the time elapsed since its beginning. The experimental protocol anticipates several epochs before the CT, but they are important only to control the experiments and the presence of relevant signals is less evident in the corresponding part of the recordings. Therefore, the beginning of each recording set is removed, and only the epochs listed on the Figure 2.8 are examined in this study.

The data shown on the Figure 2.11 was analyzed in-depth in the studies presented in [9, 10], and the obtained results will be briefly repeated in the next section.

2.5 Cell types and information coding in the 7a parietal region

An overview of functional properties of the 7a parietal region, analyzed in details in [9] and [10], is presented in this section. The role of 7a cells in translating visual inputs into motor commands, and in hand-eyes coordination in visually or memory guided reaching is studied through the single cell and population activity modulation. Global tuning field (GTF) observed in the activity of 7a cells provides an argument to the study of target and movement direction coding and reference frames in the PPC.

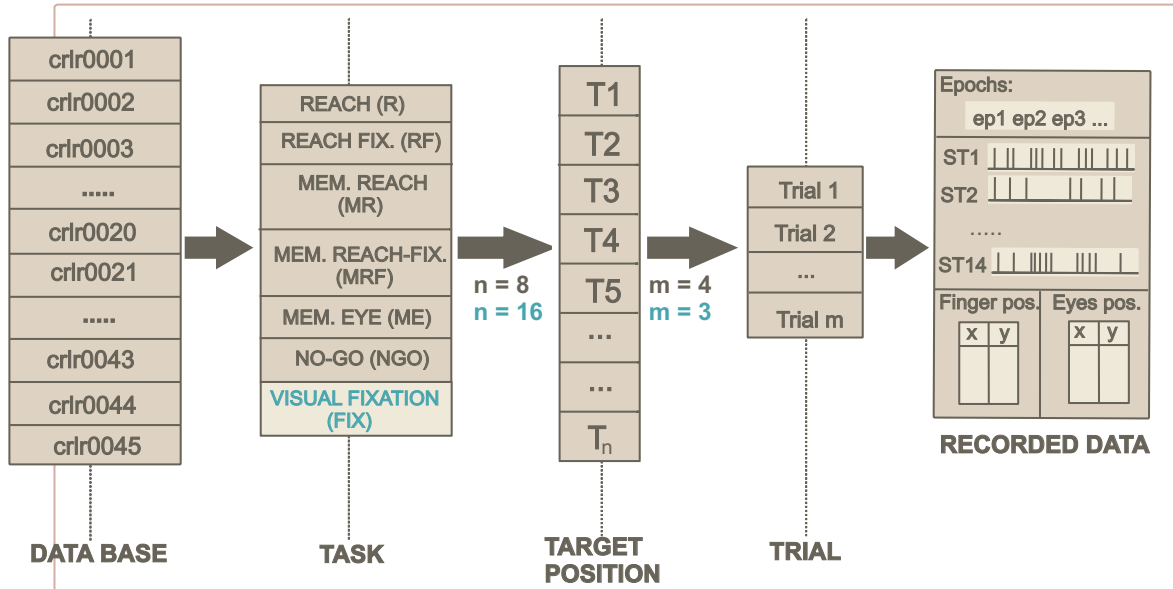


Figure 2.10: Organization of the database recorded from one of two monkeys. The first column - the database divided into files, each of them containing all the recordings collected from the same site. The recording for all seven experiments are contained in each file. R, RF, MR, MRF, ME and NGO are repeated 4 times ($m = 4$) for each of 8 target positions ($n = 8$), VFIX is repeated 3 times ($m = 3$) for each of 16 target positions ($n = 16$). The last column: all the data collected during one recording session - markers for the beginning and the end of each epoch, up to fourteen spike trains, hand and eye trajectory in 2D sampled with 10ms time step.

The significant cell activity modulation is examined through standard statistical tests (2-way ANOVA, Z-test, Rayleigh test for $P < 0.05$), depending on the particular problem of interest. For some tests, the index of modulation is calculated as the difference between the firing rate in the considered epoch and the rate in the CT (for the same trial), divided with the latter one. Also, for the study of temporal variation of the population activity, the spike density function (SDF) is calculate for each cell, using a Gaussian kernel, and all SDFs are averaged to obtain the population activity.

Neural signals: Visuospatial signals coding the target location (in the IS and Memory epochs, for example), memory of the target position, planning and execution of eye, hand and coordinated eye-hand movement (in Memory, RT and MT epochs), eye or hand positions in space (THT epoch in all tasks) are some of the signals identified in the activity of 7a cells. The activity observed in the Memory epoch, as well as the preparation for movement (except for the preparation for hand movement toward the fixated target in the MR task) are among the weakest signals observed. Modulation of the cell activity during the hand movement toward the fixated target (MTh in the R and MR tasks) is the strongest one observed in these experiments.

Visual signals: Majority of 7a cells respond to visual stimuli. This property is monitored in the VFIX task, where no memory or motor related activity is required from the monkey. First, the static target is presented in one of 16 positions, and the study of cell responses showed that more than 70% of all the recorded cells respond to the static target. In the two succeeding epochs, this target is moved from the periphery to the center, and back to the



Figure 2.11: An example of spike train recordings. The data is collected during the ninth recordings session on the monkey 'Cru'. Only one cell (the first channel) is shown on the figure for all the anticipated experimental conditions, i.e. all repetitions for all tasks and target positions. Each panel corresponds to one task. For each target position, marked on the y axis, the four obtained spike trains are shown. The short red bars mark the beginning and ending of experimental epochs. Data are aligned to the CT epoch, and marks on the figure show CT, RT, MT and THT. The x axis shows the time elapsed since the beginning of the CT. For the last task, VFIX, 48 spike trains are shown, 3 repetitions for each of 16 targets. They are, also, aligned with respect to the beginning of the control time.

initial position. A majority of the cells (more than 60%) respond to stimulus motion, some of them only to the IN motion, some to the OUT motion, and some to both of them. The influence of visual signals is also studied in the IS epoch in Memory and No-Go task. The response to the cue presentation in the three memory tasks is very similar, and always stronger than the signals found in the NGO and VFIX tasks. In other words, the reaction to a visual target is enhanced when a following movement is anticipated. The same cells are active in epochs where visual target is not present, indicating that visual and eye/hand related signals coexist in the cell activity.

Movement related signals: Planning and execution of movements is studied in the three Memory tasks, in the absence of visual signals. A majority of 67% of cells is modulated in all three tasks, and they are considered to be the 'combinatorial' cells. Spike trains are aligned to the Go-signal (i.e. the end of the Memory epoch), and the maximal activity is

compared for the three tasks. Hand-dominant cells show a peak of activity that corresponds to the preparation and execution of hand movement in the MRF task (eyes immobile, hand moves to the extrafoveal target). Similarly, eye-dominant cells are very active during planning and movement of eyes in the ME task, and much less during the corresponding epochs, related to the hand movement, in the MRF task. Finally, the eye-hand cells are equally active in all three tasks, and show a pronounced peak of activity during the RT and MT epochs in these tasks. Eye and hand related signals coexist in the majority of 7a cells, but the intensity of these signals varies for different cells.

Temporal arrangement of signals: Another interesting question, besides identification of the signals that influence cell activity, is the variation of this activity as a function of time. The study of temporal activation of cells shows that the majority of them (75%) fire after the saccade in the ME task, while a smaller subset (25%) of cells reacts before or during the eye movement. Also, the majority of cells is activated before the hand movement in MR and MRF tasks (around 60% of cells), and the remaining cells are active during and after the movement. Although this result suggests that the majority of 7a cells take part in movement planning, following a visual instruction, the temporal integration of visual and motor signals can be seen only by analyzing the population activity.

The first six tasks listed on the Figure 2.8 are considered in this study. In the Memory tasks, two peaks of the activity are observed. First of them corresponds to the visual cue presentation, indicating a strong cell response to visual signals. The second peak appears around 450ms after the Go-signal, and is similar for all three Memory tasks in spite of obvious differences between them (hand, eye and hand-eye related movements). The strongest signal is observed for coordinated hand-eyes movement. The population activity in the No-Go task results in only one peak related to the visual cue presentation, while the increase of activity after the Go-signal is lacking. The same study for the R and RF tasks revealed the two peaks of activity corresponding to those observed in the Memory tasks. Therefore, the presence of visual cue and the planning and executing a motor activity initiates an increase of population activity in the 7a cells. This supports the conclusion that 7a region plays an important role in visually guided reaching movements, combining the input visual signals with the motor signals related to both eye and hand movements.

Global tuning field (GTF) and directional modulation: Sensitivity to visual cue direction is reported in several regions of the neocortex, as already mentioned in the Section 2.2.3. The same property is observed in the 7a region, and the preferred direction is examined for various experimental conditions (tasks and epochs). The results show that 25% of the 7a cells exhibit the global tuning field (GTF) property. In other words, the preferred direction of a cell does not vary much when calculated for a large set of experimental epochs, instead, it remains confined to a small part of the 2D space. Among the examined epochs are the Memory epoch, RT and MT for all combinations of eyes and hand movements, as well as THT. This implies coordinate representation under very different conditions - the memorized target position (in Memory), coding of the eyes movement direction (in RTe, MTe), coding of the hand movement direction (RTh, MTh), coding the fixed eyes and hand position (in THT). As described in the Section 2.2.3, this involves transformation from eyes to hand-centered coordinate frame, which is not evident from this data, due to the existence of GTF. Instead, the observed GTF supports the hypothesis of the allocentric reference frame used for coding directions in the PPC (introduced in [9, 7], i.e. the coordinate system not related to neither eyes nor hand position, but to some other reference point. The question of computational properties and coordinate transform in the PPC has been extensively studied and several mechanisms are proposed in the literature [30, 60, 12, 4], [8, 7].

The directional tuning is analyzed on the population level, also. The preferred direction is calculated for all the recorded cells across different epochs. In the Memory epoch of the MR, MRF and ME tasks, also in the No-Go task, the distribution of the preferred directions is uniform in the entire 2D space. In the later stages of the reaching tasks, the preferred directions become localized and directed toward the contralateral part of the space. Additionally, analyzing the subsets of directionally tuned cells in the succeeding epochs, we can notice that only a small part of the cells remain directionally tuned in the succeeding pairs of the epochs. All the listed arguments indicate that different populations of cells become recruited in different experimental epochs.

2.6 Motivation

Basic notions related to the biological framework, necessary for further discussing this study, have been introduced in this chapter. It starts with the most general description of the mammalian neocortex and introduction of the related terminology, to continue with the posterior parietal cortex, its anatomical organization and functional roles. Several mechanisms related to the PPC are often discussed in the literature, directional tuning, coordinate transformations, intention, are also described. The particular set of experiments and the obtained database that were the starting point for this study are presented in details. Finally, the motivation behind this work will be elaborated, particularly the notion of motor intention.

The neurophysiological definition of motor intention, a high level cognitive plan for movement, is presented in the Section 2.2.3. Starting from this notion we introduced a hypothesis about its presence and absence in certain experimental epochs in the Section 2.3.3. We aim to identify this information from the neural data recorded in the PPC under the described experimental paradigm. The PPC recordings are a good starting point for the study of motor intention, since this region represents a multimodal association area, the place where sensory inputs of various modalities are integrated together to generate new high-level signals. In the context of reaching movements toward visual or memorized locations, this region plays the role of 'interface' between sensory inputs and motor outputs. The information related to the visual cue position, type of movement, directions of movement for eyes and hand, are received from the related sensory regions, and the corresponding commands are transferred to the motor areas. Therefore, the PPC contains mechanisms for initialization and early planning of movements. These are rather preliminary plans that specify information like target position or type of movement, but do not provide details for execution of those movements, e.g specific commands for muscle nerves. Instead, the PPC provides higher-level instructions, together with the sensory acquired information, and the required command is further developed in the motor and premotor cortex. This initial plan for a movement represents the motor intention, as already defined in the Section 2.2.3.

Typically, the delayed reaching tasks were employed for identification of intention in the cellular activity, as stated in the literature [4, 5]. The Figure 2.8 shows three variations of this task, delayed reaching with eyes only, with hand only, or a coordinated reaching by eyes and hand, i.e. the ME, MRF and MR tasks. The persistent activity observed in the memory delay epoch (marked as 'Memory' on the Figure 2.8) is identified as the motor intention. The arguments behind this conclusion are already listed in the Section 2.2.3, as well as the discussion of the differences between intention and other signals present in the same epoch, e.g. attention or sensory memory.

In this work we propose an alternative method for assessing motor intention. Instead of looking at the activity in the memory delay period only, we aim to test the presence of this signal across all the experimental epochs. The assumption about the appearance of motor

intention in certain epochs is already described, and will be tested through the analysis of the recorded data. At this point, some additional arguments about the epoch-intention relation should be presented.

It is difficult to relate the starting epoch, CT, to either presence or absence of motor intention, since no other instruction but to wait is given at this point. The monkey is immobile and fixates the center of the screen, but it is not sure whether it plans any future movements. Similarly, the IS is left out due to the strong cell response to the presentation of visual cue, the signal that dominates this epoch, and might 'hide' a potential movement plan. On contrary, the literature presents the evidences of motor intention in the memory delay interval, the argument that applies for the Memory epoch from the Figure 2.8. The reaction times, the preparation and planning of eye and hand movements, are straightforwardly related to the motor intention, but the argument applied for the movement times is not so evident. The MTe epoch of the R and MR tasks is likely to reflect the planning of future hand movement, while already executing the movement of the eyes. The motivation for including MTh epoch in the same class somewhat extends the definition of motor intention. These intervals correspond to the very end of the required movement, and might encode its realization rather than planning. Still, the motor related signals in the PPC are abstract representation of movements, rather than direct commands for their execution. The MTh and MTe epochs are likely to encode such high-level signals, which we still refer to as motor intention. This argument is accepted in most of this manuscript, but it will be discussed again in the context of data analysis results.

Additionally to the list of intention related epochs, it is important to select the opposite examples, those where we cannot expect to find any motor planning, at least under ideal conditions. In the first five tasks, R,RF,MR, MRF,ME, the last epoch in the protocol, THT, satisfies this condition. At this point the movement is already completed and the monkey waits for the permission to release the target. The corresponding neural activity can be related to coding of eyes or hand position in space, but should not anticipate any movement plan. The beginning 300 and the last 200 milliseconds of this epoch are removed in the analysis to account for the eventual influence of the previous movement and expectation of the next one after the end of the experiment. The NGO and VFIX tasks provide additional examples, since in both of them the eyes and hand are explicitly instructed to remain immobile for the whole duration of the experiment. The Memory and THTe epochs from the NGO are included in the analysis, similarly to the equivalent epochs from the memory tasks. From the VFIX, the three last epochs (Stat, IN, OUT) are taken into account, while CT is removed for the same reason as in the other tasks.

Evidently, a large set of relatively different epochs is included in both considered classes, the set of examples that encode motor intention, and the set of examples not related to motor intention. This approach should ensure that the only feature common to all the data in the same set is the evidence of motor intention in the cell activity. Also, it should be the only signal that makes the difference between all the examples from one and all the examples from the other set. In most of the epochs the neural activity is influenced by a whole set of coexisting signals, and it is difficult to distinguish between them when a single epoch is studied independently of the others. Instead, comparing the activity recorded in several relevant epochs, the similarities and differences between them are highlighted. For example, we can exclude the possibility to classify attention this way, since the attention should be present in the NGO and VFIX as much as in the other five tasks. Also, the data corresponding to all eight target positions are included in both classes, and coding of target positions in the neural activity should not bias the result.

Throughout this work, multiple cell recordings obtained in the same experimental session are analyzed simultaneously. The studies presented in the literature show that the intention can be observed in the single cell activity as well, but only for the appropriately designed

experimental conditions. Since, we aim to extract this information whenever it is present, it is desirable to rely on the recordings from a population of cells. For example, directional tuning of the parietal cells offers one argument for this choice. In the classical studies, the motor intention is observed in the activity of the cells whose preferred direction corresponds to the presented target direction. If we want to test for the motor intention in the entire set of allowed directions (theoretically, for every possible direction), we should be sure that at least some of the considered cells exhibit the maximum of activity for every presented direction. In other words, the population should be sufficiently large, so that the tuning properties of its cells allow good representation of the entire space. The considered data is only suboptimal in that sense, since we analyze relatively small groups of cells simultaneously. The reasons for this choice will be described in the next chapter. Also, the population size influence to the analysis precision will be further studied. Additionally, an artificial model presented in a last chapter aims to study the same question under more controllable conditions.

The main goal of this work is identification and analysis of motor intention, how precisely we can distinguish it from the data, and how consistent is the result across the epochs. Since we focus on decoding of motor related information, it is important to mention the similarities to studies carried in the context of brain-computer interfaces (BCI). The main problem assessed in these works is to extract motor related parameters from the cellular activity, that can be used as inputs for a computer or a mechanical device. Such studies aim to provide a sophisticated prosthesis for quadriplegic patients, and help them to improve the quality of life. A typical BCI assesses intact neuronal activity that can encode motor parameters, and uses it to control a computer cursor or a mechanical arm. Although studies on monkeys provided some benchmarks, there is still many technological problems that have to be solved in the future.

The focus on motor parameters in the BCI studies can be related to the work presented here, but the information of interest is somewhat different. Most of the studies reported in the literature aim to decode either the final goal position [77], or the entire hand trajectory [83], but some additional information, for example a proper timing for the go signal (see [84]), are also assessed. The approach in this study is somewhat different, since we do not aim to extract a particular set of signals from a properly chosen time interval in an experiment, but rather to analyze the properties of the neural activity during an entire natural reaching movement. The results presented in the following chapters can be of use in constructing a BCI since they can indicate how to use a set of neural recordings, which period of activity is the best for reading motor parameters, how many cells is necessary for obtaining a sufficient precision, how long a window should be used to analyze the recordings on-line, etc. The posterior parietal cortex is usually not the targeted recording site for the BCIs. Motor and premotor cortex decode motor parameters more directly, and require only quite simple, even linear [83], methods to estimate the information of interest. The parietal cortex is just a first step in generating a reaching movement, and the motor parameters are less accessible here. Still, the directional tuning is evident in the PPC cells, as well as many other motor parameters, and can be used to build BCIs [84]. The literature offers some additional arguments in favor of this approach [4]. Degradation of functional properties due to the paralysis is less apparent in the parietal than in the motor cortex. This is especially evident in the parietal regions closely related to the visual sensory cortex, due to the dependency on visual inputs that are not influenced by paralysis. The high-level cognitive signals present in the PPC are sufficiently informative for a BCI, the less explicit representation of these signals can be compensated by a more sophisticated algorithm, still sufficiently fast to work in the real time. Finally, some recent studies [16] propose recording from several different cortex regions, rather than from only one, in order to improve system robustness. Recordings collected in the PPC provide one source

of information for such BCIs. The overview of the research results related to this field, and the comparison with the work presented here will be discussed in more details in the following chapter.

Chapter 3

Machine learning based approach to motor intention identification from the experimental data.

This chapter presents the machine-learning approach to study the question described in the previous chapter. The problem statement and the relation to the brain-machine interface studies are described at the beginning, followed by the detailed description of the algorithm. The obtained results are presented in the next chapter.

3.1 Problem statement

In the previous chapter we introduced the notion of motor intention. The neurophysiologic literature refers to it as a preliminary movement plan, the first step in realization of a movement, and presents experimental evidences of neural activity related to intention [86, 87, 4]. Although the definition is very intuitive proving the presence of such a signal in the recordings is not a trivial task. Examples of sophisticated experiments designed to rule out all other signals but the intention are described in the Section 2.2.3. The activity is analyzed during the memory delay interval and the result demonstrates the presence of intention in the recordings. In this study, we aim to identify the motor intention during the entire recordings, and under various experimental conditions. In the Sections 2.3.3 and 2.6 the main hypotheses behind this study were described, particularly the relation between the experimental epochs and the motor intention.

In this section we will formulate this question as a classification problem. We are looking for a classifier able to discriminate between the segments of spike trains encoding the motor intention, from those that do not encode it. Clearly, this represents a binary classification problem, since each segment belongs to either the motor intention class or to the no motor intention class. As usual in this context, the considered database is preprocessed to obtain a set of feature vectors, suitable for describing the property of interest. Here, these feature vectors are the vectors of spike rates. One rate is computed for each cell from the considered set, where the set contains only the simultaneously recorded cells. The dimension of a feature vector corresponds to the number of cells in the set. A spike rate is the average activity of a cell, averaged over one or several appropriately chosen epochs. A label indicating one of the two classes, the motor intention or the no intention, is assigned to each of the feature vectors. It is determined according to the corresponding epoch(s), used to calculate that vector. From the classification point of view this problem is relatively simple, since it requires only a standard binary classifier, and any of the traditional methods can be employed to solve

the problem. The main difficulty is in the proper treatment of the experimental epochs. In many of them several different processes are coexisting and it is difficult to identify those which dominantly modulate neural activity. It is likely that some other signals (movement direction, eye and hand position etc) have much more influence than the one we are looking for.

It is important to point out two major assumptions behind this work. First, that motor intention can be 'read' from the average cell activity, i.e. from the spike rates. This assumption is supported by the experimental evidences presented in the literature [4], observed under well controlled conditions. For a more general situation, it is difficult to estimate the complexity of the problem, how difficult is it to extract intention from the set of various signals that play a role in performing a movement. We, also, assume that a sufficiently large population offers enough information to decode the motor intention in a general situation. The role of population coding is examined in the literature in the context of coding spatial positions and movement directions. As already explained in the section 2.2.3, and the literature listed there, the directionally sensitive cells are common in the parietal and motor cortex. The activity of those cells is described with a bell shaped curve, relatively broad, with the maximum corresponding to the preferred direction. Because a cell reacts to a broad interval of directions around the preferred one, a population of cells is required for a precise coding of spatial positions. Although the influence of the directional tuning of cells on the motor intention is less evident, it is important to take it into account. A single cell, or a small set of cells with relatively close preferred directions, is not very active for the movements far from its preferred direction(s). That cell will likely exhibit small activity modulation for all the epochs, motor intention and no intention related ones. The absence of activity in this case does not indicate the absence of motor intention in the PPC, generally. In the literature, motor intention is demonstrated only for the preferred directions of the considered cells [4]. If we want to consider this information in a general context, i.e. for every possible target position and movement direction, the population of cells is needed. Here, we included the recordings obtained for all of the eight allowed targets, and it is important that for each target at least some of the cells have their preferred directions close to it.

Additionally, the recording procedure does not allow cell selection a priori, so a set of randomly chosen cells is monitored in each experimental session. Some of the collected cells are not involved in planning and execution of movements, which effectively decreases the set size. The study presented in [9] shows a relatively small percentage of cells, 20 – 35%, modulated in two consecutive epochs, which indicates that different cell populations might be involved in different steps of a movement planning. The study [10] discusses the evidences of eyes or hand preference in the cell discharge. Therefore, low activity can indicate that the cell is not involved in control of the considered effector, that it does not contribute to a certain epoch, etc. In any case, we cannot easily relate these observations to the presence of motor intention. Also, even if the intention disappears in certain cells for a considered epoch, we are rather interested to determine whether it exist *in general* in the neural activity recorded in that epoch, which brings us back to the population coding.

Throughout this work, we analyze together only the data obtained in the same recording session. The employed measurement system contains seven electrodes, and spike trains for at most two cells can be extracted from the same recording. Therefore, the number of the cells in the set cannot be bigger than 14, and usually is between 5 and 10. The implications of the set size on the obtained results will be further discussed in the following chapter.

Finally, we should present arguments for using the selected method of data analysis. Some recent studies of neural activity recorded in the motor and premotor cortex, presented in the brain-machine interface context, show that a linear regression can be successfully employed to reconstruct movement direction or final goal from the neural recordings (see [83, 16, 50]). The

attempt to solve the problem presented here using a linear classifier was unsuccessful. Several arguments may explain this result. First, the parietal cortex encodes the motor parameters less directly than the motor and premotor cortex, since it is more distant from the regions that directly control the muscles. Also, the motor intention is a high-level cognitive process that is not easily identified in the recordings, which might require a nonlinear method to obtain it. Finally, due to the relatively small population size analyzed in this study, it is recommendable to employ a non-linear method. We will use a standard support vector machine classifier in this work, since it is proved to be a powerful tool for solving classification problems in general. Some additional arguments for using this particular method are presented later in this chapter.

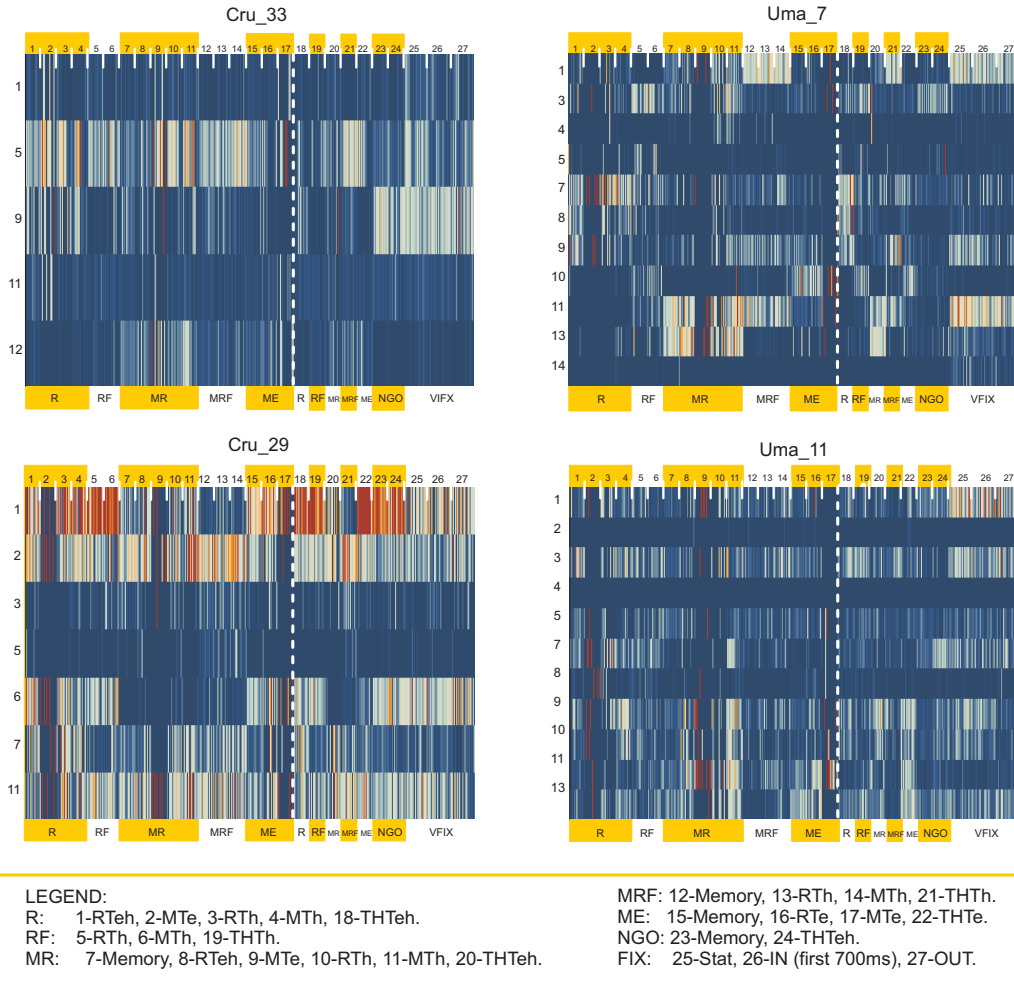


Figure 3.1: The examples of spike rate vectors calculated from the data, each vector is calculated for all the cells and one epoch. The four panels correspond to the four files, which names are indicated above the panels. The involved cell indexes are marked on the y axis, and the x correspond to the experimental epochs. The numbers above each panel indicate which vector correspond to which epoch, and the legend gives the full names of the epochs. The tasks are also marked below the panels. The spike rates are color coded, with the dark blue corresponding to small values, and dark red to the high ones. The dashed white line separating each panel is a border between the two classes.

In order to give an insight into the complexity of the analyzed problem, we show several examples of spike rate matrices in figure 3.1. Due to the normalization procedure different panels are not comparable, and from this figure we can obtain only the relative activity of

cells from the same set with respect to each other. Each row on a panel corresponds to one cell, and spike rates of the cell are represented by thin vertical colored lines (minimum is blue, maximum is red). Each number (from 1 to 27) on the horizontal dimension refers to a given epoch of different tasks illustrated at the bottom of each panel. The thick dashed white line separates the two classes. It is clearly visible that the number of motor intention examples always exceeds the number of no intention ones, which is taken into account when creating the classifiers. The names of the files are given at the top of each figure.

The presented figures illustrate the motivation for using nonlinear classifiers in solving the given task. In some examples we can notice the different activity related to the two classes, at least for some cells, like for the cell 9 in the file 'Cru 33'. On contrary, in some other examples it is quite difficult to find a simple relation between the presence/absence of motor intention and the mean spike rate, like for the file 'Cru 29'. The same examples can be used to drive some preliminary conclusions about a single cell influence on the classification performance. Looking at the average firing of the cells 5 and 9 in 'Cru 33' we can clearly distinguish between No-Go and Visual Fixation examples on one side, and all the other examples on another side. These two tasks represent more than a half of all the examples in the second class. Therefore, we can expect that each of the two cells significantly influences the classification performance. This does not hold for the other presented examples where a single cell is never sufficient for distinguishing between the two classes. We can conclude that the information of interest is encoded in a complex way, and that more than one cell of a population has to be taken into account.

Figure 3.1 also enables some preliminary analysis of differences/similarities between the different epochs and tasks. The examples calculated using the same epoch are often similar, although some fluctuations are present. Much bigger differences are detectable between the examples coming from different epochs within the same task, still we can make some general observations. No-Go and Visual Fixation examples are often similar ('Cru 33', 'Cru 29'), but in some cases the examples from the two tasks can be visibly different among themselves ('Uma 7'). No intention examples computed from THT epochs in Reach, Reach-Fixation and Memory tasks often look alike their motor intention counterparts. The motor intention examples can look very differently for different epochs, for example Memory Reach-Fixation (MRF) and Memory Reach (MR) for 'Uma 7'. Actually MRF examples are much more similar to those coming from Visual Fixation, for this file.

The classification errors obtained for these examples will be given later, together with the other results obtained from the data analysis methods presented in this chapter.

3.2 Overview of brain-computer interface studies

As described in the previous section, the problem analyzed throughout this work is related to identification of motor parameters from cortical recordings, and study of algorithms that can achieve such a goal. Similar questions are extensively studied within the framework of brain-computer interface (BCI), and the most important conclusions are listed in this section. Some main directions and problems will be described at the beginning, and latter, we will focus on the BCI paradigm most closely related to this work, i.e. the methods that rely on invasive neural recordings, particularly spike trains recordings.

In general, BCI stands for the systems that record cortex activity, apply a suitable algorithm for its analysis, and generate control signals for manipulation of actuators, like computer cursor, robot arm, or even the subject's own arm. A successful system is able to 'read' motor commands directly from the brain and performs selected movements not involving the rest of the body. Various types of movement parameters can be extracted from the recordings

and used for control. Most often, these methods aim to reconstruct hand position in reaching tasks, either the entire trajectory of a movement toward one of the possible targets [83], or to identify the target position among the finite set of possible positions [77]. Also, movement velocity is considered in some works [16], as well as motor intention [79].

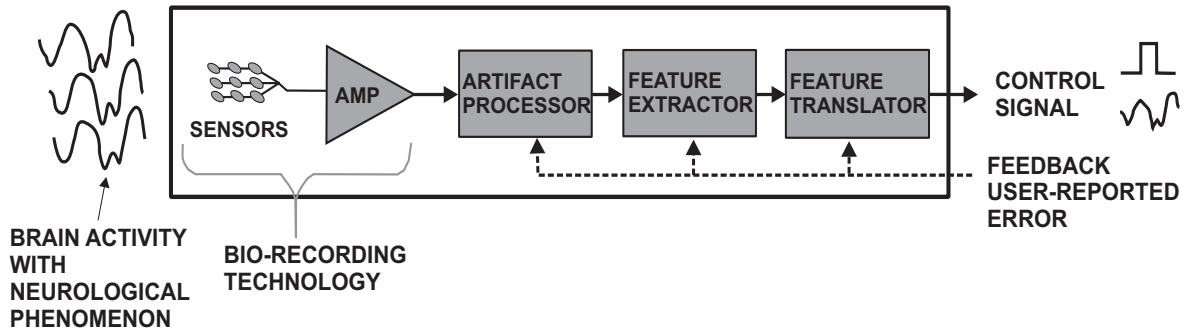


Figure 3.2: The schematic representation a BCI. The brain activity is measured and used as an input into the BCI. Sensors and amplifiers constitute the recording system. The algorithm translating recordings into control signals is divided into three parts - preprocessing to remove artifacts, extracting of relevant features from the recordings, and translating those features into the control signals (shown at the output). Additionally, in the closed-loop setting, a feedback from the user is available (shown with the dashed line).

Figure adapted from: S.G. Mason, A. Bashashati, M. Fatourehchi, K. F. Navarro, and G.E. Birch. A Comprehensive Survey of Brain Interface Technology Designs. *Annals of Biomedical Engineering*. Vol. 35, No. 2. (February 2007).

The part of the BCI systems, used for acquisition and analysis of recordings is illustrated on the figure 3.2. A set of sensors and amplifiers are employed for collecting certain types of neuronal signals, the input to the system. Generally, the applied algorithm can have three steps, the preprocessing for removing artifacts (if needed), the feature extracting from the recordings, and the method to translate these features into a control signal. Systems working in the closed-loop setting, profit from subjects' feedback when generating the control signals (dashed line on the figure). These control outputs can be used for operating a cursor, or a robot arm, for example, while feedback can be visual information about cursor/robot arm distance from the desired target.

This framework is suitable for studying functional properties of the cortical network involved in control of motor actions. The same problem has been considered by experimentalists for many years, and many questions are still open. The standard approach for assessing these questions is described in the previous section. After collecting spike trains from single neurons, the modulation of their instantaneous rates is considered in order to observe increasing activity correlated to a certain behavior of interest. The experiments were carefully designed to emphasize relevant signals. Several important conclusion, obtained through these studies, are used as a starting point for most BCIs. The directional tuning in cells of motor, premotor, parietal and some other regions, and the population coding algorithm [33, 34] are a starting point for all BCIs based on spike train recordings. Although it is known which regions take part in translating stimulus inputs into motor responses, the mechanisms of information transfer between these regions are still not entirely explained [41, 7]. The BCIs based on recordings in motor, premotor and parietal cortex has been described in the literature [39, 83, 91, 65, 84], and recently, recording from multiple sites involved in generation of motor signals is proposed [16]. This way, multiple cortical regions can be monitored simultaneously (at least a small sample of cells in each of the regions) during the planning and execution of a reaching or

grasping task. Additionally, the role of learning in adaptation to new experimental paradigm, the closed-loop tests, where the subject receives feedback signals about the motion through visual system, and other related mechanisms can be studied.

Several practical applications of BCIs are also proposed. One of them is BCI for cortical prosthesis for motor impaired patients, paralyzed due to head or spinal cord injury, including paraplegic, quadriplegic, or completely paralyzed patients, that cannot breath or make eye movements. The results, presented so far, clearly show that a variety of neural signals can serve as a basis for BCIs, but several technological issues have to be resolved before these systems become suitable as prosthesis. Some important problems, identified so far, will be discussed later in this section. Another possible use is for enhancing motor abilities in healthy humans through non-invasive BCIs, based on electroencephalography (EEG) recordings. This is the most intensively studied technology. We will describe only the basic facts related to the EEG based BCIs, together with some other types of recordings that can be employed in this context.

In the this section, several important aspects of this technology are discussed. The BCI based on electroencephalography (EEG), electrocorticography (ECoG) and some other methods for monitoring neural activity are described in the Section 3.2.1. The rest of the overview is dedicated to the invasive BCI, the approach comparable to the work presented in this study. Some basic properties of the recording methods, typical cell populations, their optimal number and characteristics are discussed in the Section 3.2.2. Several algorithms proposed for analysis of the spike train recordings are given in the section 3.2.3, along with the most important results and conclusions.

3.2.1 Noninvasive BCI and other alternatives to spike trains

The performance and the applied feature extraction algorithms are dependent on the type of recorded signals, the inputs to BCI. According to the type of sensors and signals all of the methods can be divided into invasive and non-invasive ones. We will focus on the BCI using similar data as the one considered throughout this work, i.e. the spike train recordings, and the other proposed solutions will be only mentioned for completeness.

The most common approach is to use the electroencephalographic (EEG) recordings as system inputs; according to the literature (see [58]), 83% of all the developed solutions belong to this class. The brain electrical activity is collected at the top of the skull of a subject, which provides very low spatial resolution, since each electrode records the average activity of a large populations of neighboring cells. Unlike the microwire implants, described in the previous chapter, it cannot infer the activity of a single cell. The units placed deeper in the cortex contribute less than those near the skull. On the other hand, placement of electrodes does not carry any risk, and they can cover the entire skull for simultaneous monitoring the entire cortex activity. Additionally to small spatial resolution, the neural activity is low pass filtered while passing through the skull. The entire frequency band of EEG signals consists of the following subbands, δ band for slow waves with up to 3Hz frequency, θ (4-7Hz), α (8-12Hz), β (12-30Hz) present for active, alerted subjects, and γ (26-100Hz) related to certain cognitive and motor functions. A type of waves that belongs to the α band and appears in the somatosensory cortex, the μ waves, is also relevant in this context.

The phenomenon called event-related desynchronization (ERD) is proved to be a good basis for BCI applications [3]. During an actual or imagined movement, both, β and μ signals become weaker, and the variation of the EEG amplitude in these frequency bands can be used to control a computer cursor. Typically humans subjects needed between 10 and 24 minutes to learn to adjust the vertical position of a cursor that is moving in the horizontal direction at a constant rate. After additional training they become able to use β and μ rhythms of

both hemispheres to control the cursor in 2D. This enables performing simple operations, like selecting an object on the screen. Another reported BCI is using P300 component of the event related potential (ERP) to select a cell in a 6x6 matrix [27]. The ERP is a signal that appears as a response to the stimulus of interest, and reflects the activity of many related processes in the brain. A type of ERP that can be observed in EEG recordings is P300, the response to an unpredictable stimulus (the oddball paradigm). For example, when a large majority of items that a subject tracks on the screen belongs to one category and the rest to another category, the P300 appears as a response to the occurrences of those rare items. The name indicates that the ERP typically appears 300ms after the stimulus. In the experiments described in [27], a subject is watching at the 6x6 matrix of letters and 1-word phrases on the computer screen, and focuses on one cell in the matrix. Rows and columns of the matrix are flashed randomly, and flashes of those that contain the chosen cell become relevant events that invoke P300. Detection of these signals enables selecting the right cell.

Another EEG phenomenon used for BCI systems are slow cortical potentials, visual or audio evoked potentials, and steady-state visual evoked potentials [58]. The EEG, as a non-invasive technique, is more available (and less costly) in the study phase, and also more acceptable for the patients and future applications. This particularly holds for the temporary use (in rehabilitation, for example), or for augmentation of normal body functions. Unfortunately, the sole nature of EEG recordings imposes limited utility of this type of signals. As already mentioned, the method provides a low spatial resolution of brain activity, the high pass components are removed as the signal passes through the tissue, and various artifacts influence it, like electromyographic, electrooculographic and mechanical artifacts. The EEG signals are proved to be useful in designing systems for computer-aided communication, or limited motor control. Still, the current solutions might not be sufficient for a real-time control of more complex prosthesis with several degrees of freedom [50].

One recent example of EEG based BCI is presented in [59], where the 6 human subjects were trained to perform a combination of reach and selection tasks. First, they were required to move a cursor from the center of the screen to one of the 4 peripheral targets. Then, the target should have been selected in order to complete the task. The obtained precision was 59 – 88% of correct reaching, and 71 – 91% of correct selection. The training procedure was executed in three steps, the control of vertical position only, the control of horizontal position only, and the entire 2D movement.

In terms of the spatial resolution and the size of the monitored region, electrocorticography (ECoG) offers a compromise between the microwires and the EEG recordings. A grid of electrodes is placed intracranially at the surface of the cortex, where each electrode records extracellular potentials of a small population of cells proximal to the electrode. A population related to an electrode is smaller than in the case of EEG, and the signals are not subject to distortion due to the passage through tissues and bones. Also, the higher frequencies are present in the signal. Therefore, ECoG can better capture local behavior from a certain region, although it still measures population activity and cannot assess single cells. The drawback is that it cannot sample signals from the entire cortex, as EEG, and still requires surgical implantation of sensors, although less invasive than in the case of microwires. The method is less appropriate for study on human subjects than EEG, and it was tested only on the epileptic patients during the period of preparation for the surgery. Feature extracting algorithms for ECoG based BCI are similar as those used for EEG, but here, signals are stronger and less prone to the influence of artifacts, and human subjects can learn to control them much faster [3].

Some examples of ECoG controlled BCI are presented in [51] and [28]. In the first of these studies, four epileptic patients are implanted with sensors in the left frontal-parietal-temporal region and parts of the sensorimotor cortex, which resulted in 32 recorded signals. The 1D

and 2D (four targets) control of a cursor is demonstrated using these signals. In the second report, four human subjects were trained to control the vertical position of a computer cursor and reach for one of the targets placed along a vertical line. The number of targets was 2,3,4,6 and 8. The 2 target tests gave the precision of around 70%. The only reported test with 8 targets was obtained for the generally most successful subject, and resulted in 84.6% precision.

Another recording methods that can capture neural activity have also been tested in the BCI studies, for example, functional magnetic resonance imaging (fMRI), functional near infrared imaging (fNIR), and magnetoencephalography (MEG), [3]. The first two methods measure the blood flow and oxygen consumption reflecting the increasing activity of neural cells in certain brain region. Similarly to EEG, they are not invasive and are able to record the entire brain activity simultaneously, but the oxygen flow is typically slower than the neural signals transmission and these methods introduce an additional delay in BCI functioning. The MEG measures magnetic field resulting from the electrical activity in the brain. Both MEG and fMRI require highly controlled environments.

3.2.2 Recording spike trains for the BCI applications

Most of the methods from this class use the recording systems similar to the one described in the section 2.3.1. The implanted microwires, are 20 to 50 micrometers in diameter, made of stainless steel, glass, or tungsten, and insulated with teflon or polyimide. The distance between neighboring electrodes is between 100 and 300 micrometers [81]. The recording system is equipped with a micromanipulator for fine tuning of electrode positions, in order to obtain clear and strong signals from single cells. The insertion and position tuning is very slow (100 microns per minute, for example), in order to minimize tissue damage. As described in the section 2.3.1, an on-line spike sorting algorithm for identification of single cells is a part of the system, and can be used as a feedback for tuning the electrode position. The optimal depth of implants is less than 2 millimeters below the surface of the cortex.

The number of implanted electrodes varies in different studies. The data set analyzed throughout this work is collected using a 7-electrode system. Arrays containing somewhat bigger number of microwires, 16 to 64 of them, are also used in some studies [100, 16, 68, 17]. The electrodes can be placed in a one-dimensional array, as for the system described in the section 2.3.1, or in a matrix, for example 64 electrodes arranged in 4 arrays of 16 electrodes [81]. The microwires are attached to a small connector that is fixed to the head bones typically using dental acrylic. The sufficient number of cells for a robust BCI operating in real-time is still an open question, and studies of large populations are of great interest. Properties of a high-density array of 384 channels, that can record up to 1536 units, are examined in [68]. Three arrays, each consisting of 128 microwires are connected and their acquisition systems are synchronized in order to simultaneously record large populations. Several such systems are implanted together in different cortex regions.

Alternative recording device are microprobes built on a silicon substrate. Two such systems are developed at University of Michigan and University of Utah, and described in [81]. The electrodes are cut into the surface of a silicon plate and shaped into needles with similar tip as for microwires. Unlike them, these arrays are not fixed to the skull but have certain flexibility and can 'float' of the surface of the cortex. Such systems allow cortex activity recording using a large density of electrodes per surface unit, but, it is not possible to tune the position of each electrode separately. The Utah array consists of 10x10 electrodes, where 96 out of 100 are available for recording. The total surface of the platform is 4x4 millimeters, with the distance between electrodes around 400 micrometers. Its properties are extensively examined in [89].

Invasive recordings inevitably lead to the tissue damage. Placement of the electrodes causes the risk of inflammatory reactions, as well as the vascular damage which leads to microhemorrhages. Infections caused by problems with the acrylic cap, used to fixate the connector and protect the electrodes and the tissue, are reported in [89]. The damage of neurons activates glial reaction in order to absorb affected cells and repair the scar in the tissue. The infiltration of these cells causes the change in the concentration of calcium and potassium in the electrode environment, decreasing the neural activity that can be recorded. Eventually, a thick sheet of insulation is formed around the electrodes disabling further recordings. These problems affect the quality of recorded signals in a relatively short period after the implantation. The influence of the electrode tip shape and the material they are made of is examined, the shape of the tip determines the tissue damage caused during implantation, while certain types of materials can attract cells to stay in the proximity of the electrode [81]. Some researchers [47] propose the use of Neurotrophic Electrodes that encourage the cell growth inside the hollow electrode tip. This way, the tissue is held attached to the electrode, which improves recording quality and stability.

Stability of recordings in a longer period is also a relevant issue. With the time, some of the cells around electrodes die and the signals they can collect become much weaker. This problem needs to be resolved for any reliable BCI application. While most of the studies use data collected several weeks after the surgery, recording over longer intervals is also tested. In [81], the successful recording during one year period is reported. In [68] the results obtained 30 days after the surgery and 18 months after it are compared. Thirty days after the surgery 54% of all the implanted electrodes record a waveform that enables identification of at least one cell. Eighteen months after the surgery that percentage was 35, but the statistical properties of the recorded signals did not change. Microprobes allowed successful recording over the period of 569 days, and during that time characteristics of the collected signals (the shape of the waveforms, signal to noise ration, peak to pear potential etc.) did not vary significantly [89]. The same authors suggest that this type of electrodes may provide greater reliability during long time recordings than the microwires. Characteristics of the optimal recording system have to be further tested.

The signals accessible using the devices described here are spike trains and local field potentials, and spike trains can belong to single units (i.e. isolated cells), or multiunits (several cells recorded together, without spike sorting procedure). The collected population size is limited with the number of available electrodes and the number of recording sites on the tip of electrode. In the database described in the previous chapter, the 7-electrodes system recorded between 2 and 14 cells, most often 8 to 11 of them. The study [68] shows recording of 56 cells from the 96 electrodes in one monkey, and 247 cells from 384 electrodes in the best session with the second monkey. Also, it compares the efficiency of recording in several brain regions, primary motor cortex (M1), premotor dorsal cortex (PMd), and somatosensory region S1. The best efficiency was obtained in the motor cortex, but other regions also provided reliable signals for cell identification. In [89] the average of 120 waveforms was obtained from a 96 electrode microprobe, but the single units were not separated from multiunits.

Since this chapter presents a review of issues related to BCI technology, all of the reported tests were carried in one of the cortex regions involved in movement planning or realization, the primary motor cortex, premotor cortex, supplementary motor area, the parietal cortex, even some somatosensory regions.

3.2.3 Algorithms, functions, and precision

This section gives an overview of most important methods applied for extraction of motor parameters from the recorded neural signals, and the obtained precision when these parameters

are used for controlling an actuator. The following report summarizes the major contribution of several groups working on BCI development, and the text is organized accordingly. The results particularly interesting for the comparison with our work are described in a separate section, in order to point out some major similarities and differences.

The group of Andrew B. Schwartz extensively studied the algorithms for estimation of 3D movement trajectories. Population vector coding and principal components analysis were some of the methods applied to study motor parameters in various tasks. The first of them is introduced in [34, 32] as a mechanism of movement direction coding in the motor cortex. Assuming a sufficiently large population where the preferred directions of the involved cells uniformly span the entire space, a linear combination of their instantaneous firing rates gives a good estimation of an intended movement direction. A more detailed description of the model is given in the section 2.2.3. Successful implementations of the population vector coding algorithm in the 3D reaching tasks are reported in [93, 92, 82, 98]. In these studies the subject (rhesus monkey) is required to reach for one of eight targets on the vertices of a cube in the virtual environment. Relatively small populations of 30 to 40 cells are used most of the time, and the performance of only 18 cells is reported in [91]. So small population size is used to emphasize the importance of visual feedback in controlling hand movements. The correct target position prediction increased from $(27 \pm 9)\%$ when the visual feedback is not available, to $(49 \pm 17)\%$ when it does exist. Note that with 8 target positions, completely random prediction would lead to around 12% accuracy.

Alternatively, the algorithm based on principal component analysis is tested in the same environment [39]. The instantaneous firing rates of 30 cells are used to calculate movement representations in the principal components space. The new data segments are matched to these representations and the corresponding velocities were assigned and used to calculate movement trajectories. The obtained precision in estimating a final hand position is 68.5% in average, and in the best case 87.5%.

Reconstruction of a more complex trajectory using the population vector coding is shown in [80]. A monkey is required to perform a constrained hand movement, i.e. draws a line presented on the screen. The successful estimation of a spiral trajectory from the population of 349 cells was demonstrated. The cells were recorded sequentially from the arm region in the motor cortex. Similarly in [82] the representation of drawn trajectory in a 3D virtual environment is studied. The neural activity is recorded from the motor and premotor ventral regions (PMv), and the task involved an optical illusion in order to disassociate actual and visualized movement representation. The activity of the motor cortex describes well actual hand movements, while the PMv one corresponds to the planned/viewed trajectory. Both of the regions are involved in movement planning, but only the PMv receives a feedback about the completed action.

The information transfer rate during a task execution was studied in [92]. A method to control efficiency-error rate trade off is proposed, based on the fact that the majority of information is transmitted in the initial phase of a movement. If a certain error rate is allowed, a small initial interval of neural activity may be sufficient for operating a BCI.

In addition to the cursor control tasks, several studies tested the control of a robot arm [93, 92, 98]. The monkeys were involved in either standard center-out reaching tasks in 3D, or in self-feeding tasks where the food position was randomly placed in the working space. The obtained precision in a 30-40 minutes session in the self-feeding trials, reported in [93] was around 70% for a robot arm of six degrees of freedom. A recent study [98] examines control of a robot arm with a gripping device attached to it. In a series of trials, the monkey was required to complete an entire movement, reach for the food, grab it, and deliver it to the mouth. The two tested monkeys obtained success rates of 61% and 78% respectively.

In the studies carried by the group of M. A. L. Nicolelis, linear regression models have been successfully applied for estimating movement trajectories, together with some other motor parameters, like velocity and gripping force. Large populations recorded in several brain regions are proposed in order to ensure reliability and long-time stability of BCI.

A study on rats [20] demonstrates the successful estimation of movement timing using principal components analysis in combination with artificial neural networks. The study revealed the importance of using spatial (number of cells) and temporal (time evolution of the activity) information when decoding motor parameters. The proposed method is further tested and compared with simpler linear regression models in [100]. No significant difference in the performance of these models was observed. Linear methods are further examined in [16], where various motor parameters, hand position and velocity in 3D, gripping force, and electromyographic activity (EMG), are estimated. The hand position was predicted with 85% precision, hand velocity with 80%, gripping force 90%, and EMG 61%. The spike rates were calculated using a window of 100 milliseconds, and 10 latest such values were combined in a linear regression.

The optimal size of a recorded population is extensively discussed in these reports. Since the standard recording systems do not allow a priori testing of cell properties, it is important to collect large populations, preferably from several cortex regions, in order to obtain stable and reliable systems. In [100] the activity from primary motor cortex (M1), premotor dorsal cortex (PMd), and posterior parietal cortex (PPC) was recorded from both hemispheres. The total number of implanted microwires was 128. A similar recording technique is proposed in [16] and [17], where data was recorded from M1, PMd, the MIP region of PPC, and supplementary motor area (SMA). Also in some monkeys the primary somatosensory cortex (S1) was monitored. Description of the technique for implanting multiple microwires in various cortex regions is discussed in details in [68]. The tests showed that all of the recorded regions influence the results, but not equally for each parameter estimation. For example, SMA predicted hand position and velocity well, but the gripping force much worse. The PPC population predicted gripping force and hand velocity well, but it was not so reliable in predicting the hand position.

The optimal population size was assessed using the neuron dropping technique, which tests how the population size influences the algorithm precision. The test was carried separately for each cortex region. In [100] the total number of cells, sufficient for 90% precision, is estimated to 400 to 1200 cells, depending on the considered region. In general, the contralateral PMd required the smallest populations, while ipsilateral M1 and PMd needed the biggest ones. Using particularly large populations increases the computational complexity of calculations. The authors, also, speculate that it might bring a risk of overfitting, since every new input adds a group of new parameters to the algorithm. The methods for prior cells selection is reported in [76]. In general, large populations are good in absence of prior cell selection, but selection of an optimal subset before classification improves characteristics of the method. Still, every such method can be applied only after the recording is done.

Adaptation of the neural activity to the performed task is reported in [16]. Cell tuning is modified and the estimation precision increases. Usually, all the model parameters were re-trained at the beginning of each daily session. In order to ensure good generalization of the developed system, it is important to test it using somewhat modified tasks, as proposed in [100].

The recent activities of this group are directed toward real-time operation of robots using the activity recorded from the monkey cortex, while performing tasks similar to those required from the robot [45].

The most interesting results, in the context of our work, are reported by the group of R.

A. Andersen. They studied the BCI development based on the activity of parietal cortex. The neural activity is recorded mainly from the parietal reach region (PRR) that consists of the MIP area and a part of the area 5. Its activity reflects the planning of hand movements. Several arguments for the PPC based neural prosthesis are listed in [84] and [65]. This region receives direct inputs from the sensory regions, including the visual ones, and it is more distant from the motor ones. In paralyzed patients, the brain regions providing a direct control of limb movements may suffer from damage, the process that affects motor and premotor cortex much more than the PPC. The PRR takes part in a movement planning, and its stable, persistent representation remains coded in the neural activity, no matter is it executed or not. It can be advantageous in developing prostheses for the paralyzed patients, since they should operate on the activity related to the imagined movements.

The target position in a 2D reaching task is decoded using the maximum likelihood estimation (MLE). Prior to the target decoding, the *state* is estimated. The state is used in the same context as epochs in our study. In other words, every movement is divided into several states/epochs, and some of them can be recognized from the corresponding neural activity. This approach provides a method for the automatic release of start and stop commands for a BCI, in other words, when the effector should execute a movement, and when it should wait. In many other studies this question is not assessed and the activation/deactivation of a BCI has to be done manually. In the MLE algorithm, the uniform prior probability is imposed by the task itself, and the adopted posterior distribution is Poissonian. Its parameter is estimated from the training data, for all the possible target positions and states. The spike rates are calculated using a 250 milliseconds window. A finite state automaton is added to the system, and uses the inputs from the classifier in order to provide a smooth transitions between the states. Relatively small populations of well tuned cells are used in these studies, 8 in [65], or 23 and 41 in the two subjects in [84]. The classification precision demonstrated in the first of these reports is 64.4% and 63.6% when 4 and 6 targets are available, respectively. In the second one, the precision of nearly 90% is obtained using 40 neurons and 8 targets. These results might indicate that the additional state estimation procedure stabilizes the prediction of hand movement, so even small populations are sufficient for obtaining high precision.

The question of state estimation will be further discussed in a separate section, and the comparison with motor intention decoding, presented in this work, will be emphasized.

The idea of estimating states together with target positions is further elaborated in the studies of the K. V. Shenoy’s group. In [2], a finite state automaton is implemented to enable smooth transition between the three states, the initial time, planning of a movement, and the movement realization. This system defines the timing of motor commands sent to the effector, as well as the interval used for target decoding.

The activity of M1 and PMd cortex is sampled using a 96-electrodes system, that recorded between 100 and 200 units (single- and multi-units). The same maximum likelihood method is applied for identification of states and targets. The optimal number of units used to operate a BCI is tested using the same neuron dropping technique as in [100], and [16].

Additionally, the influence of the neural activity integration time is tested. In the previously listed studies, the time interval used to calculate spike rates, i.e. the integration time of neural activity, is fixed in advance, according to the prior knowledge about the dynamics of neural processes. In [92] the proposed optimal integration time for the signals from the M1 region is 30 milliseconds, since the information transfer rate starts to saturate 30 milliseconds after the go signal. The linear models reported in [16], and [100] use the bins of 100 milliseconds to calculate spiking rates, and the 10 last bins are combined to predict the future target position. Similarly, in [83], the most recent second of activity was divided into 50 milliseconds bins for calculating a linear regression. The integration time of 250 and 500 milliseconds is

employed in [84], while [65] examines intervals even as big as 900 milliseconds, in order to obtain precise classification with only 8 cells. In [77], the influence of the integration time to the algorithm accuracy, and to the optimal information transfer rate capacity (ITRC) is assessed. For the first parameter, the relation is straightforward - the longer integration time provides better accuracy. The second measure, ITRC, is a more reliable indicator, since it examines the trade off between the precision and the time required for transmission of certain information. For a small integration time, the error rate is high but many trials can be attempted in a short time. As it increases, the error rate decreases, together with the number of trials in the time unit. The values between 10 and 600 milliseconds are examined for the number of units between 20 and 150. For a small number of units, the optimal integration time is around 120 milliseconds, and it rapidly decreases for the increasing number of units. Alternative approach to calculation of a fixed optimal integration time is presented in [2]. Here, the finite state automaton is employed for defining the optimal segment of neural activity to be used for the estimation of movement directions. The variable length of the integration time can compensate the nonstationarity of the recorded signals. Also, the trade off between the error rate and the latency can be calculated on-line for each trial. The fixed window of 200 milliseconds provides 82% accuracy, and every strategy with the adaptable length performs better in terms of the obtained accuracy, but the longer time to generate a motor command is required.

The group of J. P. Donoghue examined estimation of continuous 2D trajectories from small populations of units recorded from the M1. The number of recorded units varied from less than 10 to around 40. The successful applications of linear methods, like linear regression [83], or Kalman filter [11], are reported. In [83], a linear model controlled by neural activity accounts for more than 60% of variability in the actual continuous hand movement. A detailed study of neural activity during smooth pursuing-tracking tasks was presented in [71]. This type of movements requires constant change of movement parameters which results in the highly nonstationary neural signals. The initial position was chosen randomly on the 2D screen, and the required movement followed a pseudo-random trajectory. The dynamical tuning of cell activity to account for changes in position and velocity was assessed, showing that most of the cells encode both of the signals, but these two information items are differently weighted in different cells. The whole population contains a large spectra of these weightings. Measuring the information content in these cells showed that that both position and velocity are only weakly encoded, but the successful decoding of a continuous hand trajectory using linear models was demonstrated.

Recent reports from the same group focus on the tests on human subjects [38, 95]. Two paralyzed patients were involved in these tests, and the activity of small populations of M1 units (3-57 of them) was collected. Since the patients were not able to move their hands, the training sessions were somewhat modified with respect to the usual. The subjects were required to 'imagine' a hand movement following the cursor moving on the screen, and the parameters of a linear model were updated in the succeeding blocks of trials. The quality of BCI control was also assessed in the typical center-out tasks, and the 73 – 95% precision was reported in [38]. The question of neural tuning to position and velocity, and its variations during a continuous tracking task was assessed in [95].

Another study on a human subject is presented in [47]. A single neurotropic electrode was implanted in the human cortex and the multiunit activity was recorded. These electrodes encourage cell growth close to the electrode tip, which can provide recording of stable signals over a longer time course. The subjects were trained to operate a computer cursor and select icons, or letters from a virtual keyboard, which were used as an input to a speech synthesizer.

In the tasks where the subject was instructed to spell a given word, the obtained spelling rate was 3 letters per minute. Typically, in one session, some initial trials were wrong, but the accuracy increased over time, to drop again close to the end of a session as a result of tiredness. A similar BCI, also for human subjects, is developed in [46]. Here, the cursor position was controlled using LFP signals, instead of spike trains.

In the study reported in [73] the hand EMG activity was predicted instead of its kinematics. These types of signals are stochastic and noisy, and more difficult to predict than the movement parameters. Still, the accuracy of 75 – 80% is reported using linear models. The activity of 66 units was recorded, but only 35-40 well modulated units were used. The monkey was trained to reach for one of four buttons placed in a vertical plane. The EMG activity was predicted using a linear filter and 500 milliseconds of neural activity, the interval selected testing the values between 100 and 1000 milliseconds. Although the overall precision was good, the models demonstrated two typical errors, they failed to predict extreme peaks in the EMG activity, and they tended to predict activity in the intervals between the recordings due to the random neural firing in M1. Some additional issues, like stability and generalization of prediction, are also discussed in this report.

Finally, it is important to mention the study presented in [69], since it uses the same type of neural decoding algorithm as the one adopted in our work, i.e. the support vector machine classifier (SVM). The experiments are carried in rats, and the animal was trained to press right or left paddle in order to illuminate a LED in the center of the panel. The visual and auditory feedback was given every time the paddle was pressed, and the correctly completed tasks were rewarded. The activity of 10 neurons from the motor cortical area was collected, and their spike trains are divided into a fixed number of bins. The dimension of the obtained spike rates vector, the input into the SVM, was equal to the number of cells multiplied with the number of bins. The obtained accuracy in the closed-loop tests varied between 71.7% and 85.81%.

3.2.4 State estimation for the autonomous BCI

The reports on state estimation from the parietal cortex activity, presented in [84, 79] and [65], and the premotor cortex activity, in [2], are additionally discussed in this section, and compared to our motor intention identification task. As previously described, these studies aim to decode the states of a typical reaching task, together with identification of the intended target position. This approach automatically generates 'go' and 'stop' signals for the BCI, and also improves the choice of optimal interval of activity for the target identification task.

Tasks: The experimental setup in these studies is very similar to the one described in the Section 2.3.3. The monkey was trained to reach for a peripheral target in the 2D workspace. In the initial epoch, the Baseline state, no activity is allowed until a new instruction signal arrives. Approximately 500 milliseconds later, the target position is presented, and its disappearance indicates the start of the Plan state. The last relevant interval, Reach, is the period between presentation of the go signal on the screen, and the start of the hand movement. During the entire trial, the monkey was required to fixate the center of the screen with its eyes. This experiment corresponds to the Memory Reach Fixation (MRF) task, described in the Section 2.3.3. The Baseline is equivalent to the Control Time (CT), Plan state to the Memory delay interval (Memory), and Reach to the Reaction Time for hand (RTh).

Duration of these epochs may vary significantly from the values in our study, given on (see 2.9). In [84], the Baseline is around 500 milliseconds long, while CT lasts between 1000 and 1500 milliseconds. The target is always presented in a fixed interval of 300 milliseconds.

The Plan state is around 800 milliseconds long, while our Memory interval lasts between 1.5 and 2 seconds. In [65], the epoch durations are similar to the ones in our tests (Plan state is between 1.2 and 1.8 seconds long). The 'Reach Trial' described there corresponds to our MRF task. In the additional task, the Brain Control, a cursor controlled by the neural activity reached for the target instead of hand. This kind of tasks, very important in the BCI context, were not anticipated in our study, since its primary goal was analysis of the signals in the parietal cortex and assessing the mechanisms of hand-eyes coordination, rather than building an operating BCI. From that point of view, our study assesses the information coding of motor parameters in the cortical signals, without aiming to develop an operating BCI. The study in [79], included the delayed arm and delayed saccade task, i.e. the equivalent of our MRF and ME tasks.

Finally, the available number of trials for each of the tasks, should be compared. In [65], each daily session consisted of 250 to 1100 trials. The first segment, used to train the algorithm, included 30 trials in each of the 8 directions. In [84], 8-16 trials per task per direction was performed (see [86]). In our database each task for each target position is repeated 4 times, except the VFIX task which has 3 repetitions. The smaller number of repetitions is compensated by analyzing different task epochs separately, so that each trial gives more than one vector of spike rates.

Recording sites: Three among the listed studies, [84, 79] and [65], record from the posterior parietal cortex, precisely, from the region named parietal reach region and include the MIP and a part of the area 5. As already described in the Section 2.3.2, our database was collected from the electrodes implanted in the area 7 of the PPCs. Activity of the cells in the PRR is dominantly modulated by hand movements and hand positions. On contrary, the area 7 activity is believed to be dominantly modulated by saccadic movement, although the study in [10] revealed the presence of hand dominant and combinatorial cells, together with the eye dominant ones.

Number of units and the integration time: The number of monitored cells differs throughout these studies, but, in general, relatively small populations are used in all of them. The [65] examines movement decoding from a very small set of 8 cells, but requires 900 milliseconds of the neural activity in the memory delay epoch for the target prediction with around 64% precision. In [84] the populations of 23 and 41 neurons is examined, and the time used to calculate spike rates is 500 milliseconds (and 250 milliseconds in some cases). The precision obtained using the 'time transition' rule, should be emphasized here, since it can be compared to our results. The percent of correct identifications of Plan activity, using the 40 cells almost reaches 100% precision, but it decreases with the number of cells. For 10 of them, the precision is around 85%, and for 20 around 90%. In [2], between 100 and 200 units (mainly multi-units) were recorded from the M1 and PMd cortex. The algorithms proposed here require a certain fixed number of Plan state classifications from the data, before releasing a 'go' signal. Precision of the Plan state estimation is not given directly, but through the latency estimation, i.e. the time needed to obtain the required number of correct Plan classifications and generates the 'go' signal. If this number is set to 23, the average latency is 350 milliseconds. Finally, the study in [79] examines the movement intention identification from the spike trains and local field potentials simultaneously. The precision of 90% can be obtained using 20 channels of LFP activity, but the spike trains from 50 units are required for the same result. Here, it should be noted that the signals combined in the same decoding session were not, necessarily, recorded simultaneously.

Our database, described in the Section 2.4, is also recorded during many sessions, but only spike trains obtained simultaneously were analyzed together. This limits the number of

available cells to 2-14.

Decoding algorithms: The data analysis method differs between our study and the literature. All of the listed papers use the maximal likelihood estimation, with the uniform prior probability, and the Poissonian posterior. The parameters of the posterior distribution are calculated from the data. One model was required for each target and state, and the new data was compared with each of these models. Our approach applies the standard support vector machines (SVM) based classifier. This algorithm does not depend on the number of states or targets, and can be generalized easier, for example, if the target can appear in any random position in the working space. The analyzed data is assumed to be stationary in both, our work and the studies presented in the literature. For developing the MLE, all spike trains are assumed to have Poissonian distribution. Our approach does not require hypothesis related to the data distribution, but estimates instantaneous firing rates using finite length intervals, even moderate size intervals. We assume that an instantaneous spike rate does not vary significantly within one interval. Therefore, the stationarity is assumed although not explicitly verified, but it is acceptable when using sufficiently short intervals. All of the parameters are calculated with respect to the experimental epochs, and we assume that the neural activity does not vary much within one epoch. If the variability was expected at the beginning or at the end of an epoch, the corresponding part of the recordings was removed before calculating the rates. The stationarity between different trials of the same experiment is also assumed without verifying it.

Additionally to the arguments already listed, the main difference between our study and the references is in the sole definition of motor intention. Instead of classifying the experimental epochs using their distinct neural activity, we aim to find the similarities between certain selected epochs. In other words, to identify a signal interpreted as motor intention in all of the epochs where it should exist. In order to be sure that the classified information really represents motor intention, variations of the reaching task and several epoch in each of them are included. According to the listed literature, the experimental epochs are sufficiently different among themselves, and can be reliably distinguished using a standard machine learning method as MLE. This idea can be anticipated from the Figure 3.1, where some of the epochs are visibly different from the others. Still, identification of the information of interest, present in all of them, and having in mind the demonstrated variability among them, is not an easy task.

In the context of the BCI design, recognizing single states is clearly sufficient for building a fully automatic system. The motor intention signal can represent an alternative to the state estimation. Although we use the entire length of recordings to train the classifier, it is not necessary in the test phase. There is no obstacles to test only the intervals before the actual movement, when generating go signals and estimating target positions. Theoretically, the motor intention can be more useful for estimating the control signals of a continuously moving BCI. This type of analysis can be used to identify the beginning and ending of a motor activity, even for the movement epochs differently defined. Also, it does not impose limitations related to target positions in the workspace. Unfortunately, motor intention extraction from the data collected during continuous movements cannot be tested within the framework of this project.

The report in [79] proposes using the LFP activity instead of spike trains for movement intention identification. The presented results indicate that spike trains are more suitable for target decoding, but the behavioral states appear to be better represented in the LFP activity. The comparison between different signals representing neural activity was, also, not possible in our study, since the only available database contains exclusively the single-unit activity.

3.3 Data Analysis Methods

In the introductory section of this chapter we elaborated the motivation behind this study, along with observations from the mere visual inspection of the data. Its goal was to develop some intuition about the problem of interest, and indicate the specific issues and difficulties that could emerge while investigating this problem. An illustration of a typical set of spike rate vectors, calculated from the data, is given in the Figure 3.1. In the classification context, this is a set of feature vectors representing the data, used as inputs into the classifier. Here, we will describe the methods for computing these feature vectors, the classification algorithm, and the details of its tuning, necessary for solving this particular task. Two variations of the analysis are tested, and the results will be described in the next chapter.

The section is organized as follows, first, the methods for preprocessing the database of spike trains is described in the Section 3.3.1. A general description of the support vector machines, included for completeness, is given in the Section 3.4. Finally, details of the algorithm implementation and tuning is given in the Section 3.4.2.

3.3.1 Two Approaches to Feature Extraction From Spike Trains

In the neurophysiologic literature [4], motor intention is identified by examining the modulation of neural firing at a time of movement planning. Motor intention is observed as a persistent increase of neural activity in the memory delay epoch, i.e. in the period between the presentation of the target to be reached and the beginning of the movement. The algorithms for state identification, described in [84, 65, 2], also analyze cell activity on the level of spike rates.

We adopted the same hypothesis, namely, the presence of motor intention can be described by spike rates, the mean frequency of spiking calculated within the selected interval of time. Therefore, in the first algorithm step, a set of spike trains is converted into several multi-dimensional vectors of rates, where each 'dimension' corresponds to one monitored cell. Only the data recorded in a single experimental session is combined in the same decoding task, which limits the population size to up to 14 cells, as well as the maximal dimension of rate vectors. Some reports [80, 78] adopted the alternative approach combining the activity recorded during different sessions to obtain populations of several hundreds of cells. Evidently, a real-time operating BCI cannot use data collected sequentially, but these studies are useful for preliminary tests of the algorithms. In the studies where neural decoding was tested on-line, up to 200 units were employed, although lot of effort is directed toward developing systems that can record more than that. For example, a device that can (theoretically) record more than 1500 cells simultaneously was examined in [68]. In this work we decided not to combine data from different sessions. It is not entirely clear which processes influence the activity of the monitored cells, and some of them might vary significantly on a day-to-day basis. Motor intention, as a cognitive signal, is not straightforward to estimate and may be additionally tainted by these effects. In favor of this assumption, several reports emphasize the importance of tuning algorithm parameters at the beginning of every daily session, in order to adapt to unpredictable changes in neural activity [16, 17]. We believed that the data recorded simultaneously can show the signal of interest more clearly, in spite of being limited by a small population size.

Additionally, combining different sessions will disable testing some parameters of interest, for example the optimal window for calculating spike rates. The preformed tasks do not have a fixed duration (except for the period of visual cue presentation), and the length of certain epochs vary in different sessions. Therefore, using the fixed window size disables tracking the epochs that correspond to the intervals of neural activity delimited by the window. An

estimation of the algorithm performance for bigger populations is assessed through an artificial model, described in one of the following chapters.

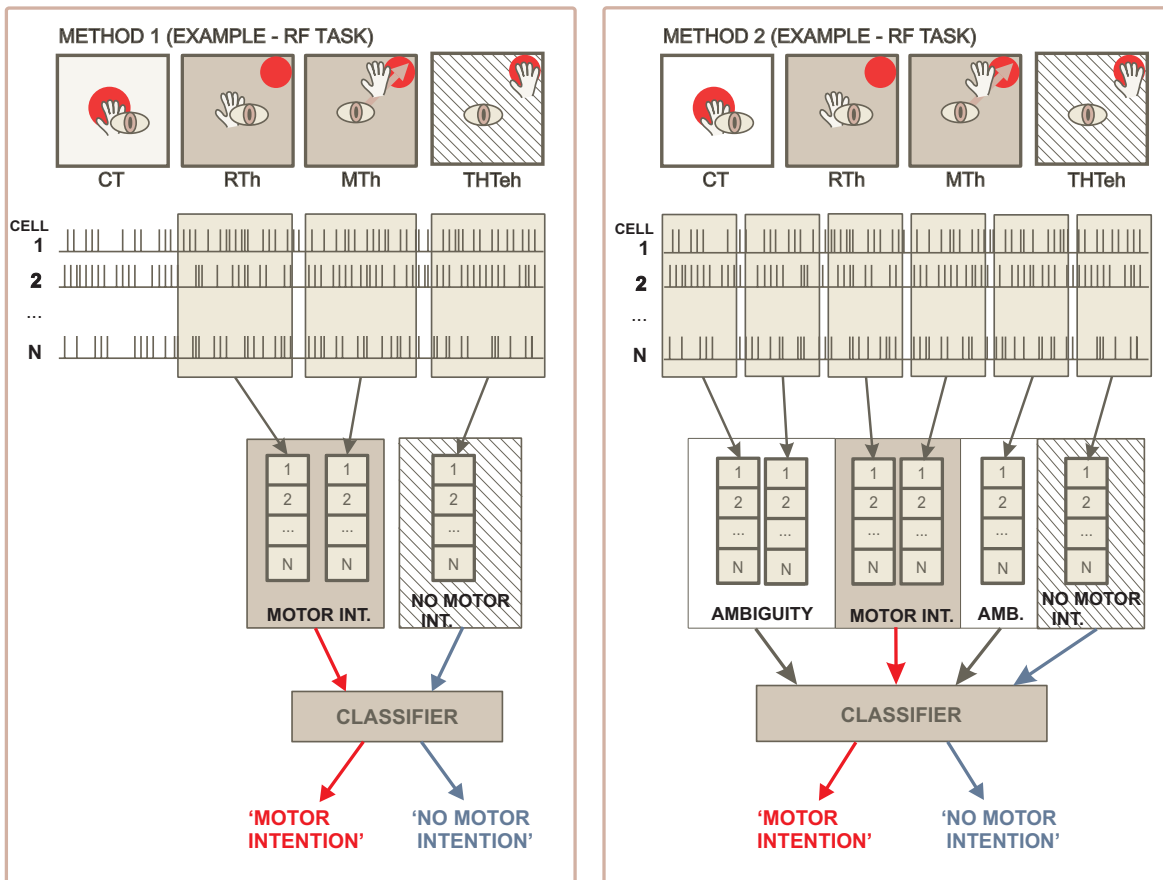


Figure 3.3: The schematic representation of the data analysis procedures. Top - an example of an experiment is presented. The collected spike trains and shown below, together with the two methods for calculating spike rates. The obtained rate vectors are assigned to the two classes, in accordance to the epochs. Some of the vectors from the right panel cannot be labeled since the presence/absence of motor intention is not clear in the corresponding epochs. Bottom - a classifier selecting between the motor intention and the no intention is trained and tested using this data.

A schematic representation of the steps in data analysis is given on the Figure 3.3. Both methods are illustrated for the Reach Fixation task, but the same procedure is repeated for each of the tasks from the Figure 2.8. A collection of N spike trains recorded during this task is also shown, where N can be between 2 and 14. Spike rates are calculated out of this data using windows whose size and position are defined by the two methods from the figure. They will be described in details in what follows. For each window position, a vector of spike rates is calculated by evaluating the average spiking frequency within the window for each of the N cells in the set. As usual in the machine learning problems, labels are assigned to the obtained feature vectors. They are divided into the motor intention and the no motor intention class, according to the position of the window with respect to the experimental epochs. Presence or absence of motor intention in certain epochs is already discussed in the Section 2.3.3, and illustrated on the Figure 3.3. As before, the shaded epochs should exhibit motor intention, the hatched ones its absence, and the white ones are ambiguous and we can

not determine which class they belong to. The same 'code' is used for marking the classes of rate vectors. If the window covers only the segment of spike trains recorded during the motor intention epochs, the resulting vector is assigned to 'motor intention' class. The same holds for the opposite class. The calculated feature vector is labeled as 'no motor intention' if the corresponding window overlaps only the no intention epochs. In case when a window mixes two epochs of the opposite types, or uses part of the ambiguous ones, the resulting feature vector cannot be labeled. All the labeled vectors are divided into the training and the test set, and used to develop an SVM classifier and examine its performance. The vectors that cannot be assigned to any of the classes are used only in the test phase. Once trained, the system is able to recognize which rate vectors encode motor intention and which not. It is assessing a question how reliably the motor intention can be estimated from the data, and which parameters influence the performance in solving such a task.

Epochs classification

In the first method (left panel on the Figure 3.3), we aim to classify the experimental epochs according to their relation to motor intention. Knowing the key processes related to each of the epochs, we can straightforwardly assign presence or absence of motor intention to some of them, as shown on the Figure 2.8. Naturally, the behavior of the monkeys is not so well determined, especially in certain experimental phases; therefore some ambiguity about the classifier output is always present. The motive behind this approach was to test the aptitude of a sophisticated algorithm to identify this information in each epoch. We assume that the average activity of a small set of cells is sufficiently informative for such a task. This analysis allows us to test the initial assumptions regarding the considered problem, and also to examine the limitations imposed by the experimental design, to select the appropriate analysis algorithm and its optimal parameters.

Spike rate vectors: Before calculating spike rates, the notation for epoch marks and spike trains have to be introduced. The number of epochs depends on the task, and even for the same task their duration changes in different trials. There are 6 epochs for the R, MRF and ME tasks, 4 for the RF, NGO and VFIX, and 8 for MR task. The number of trials, repetitions of the same task for the same target, is 4 for all the tasks except VFIX where it is 3. The available databases are illustrated on the Figure 2.10. The rightmost block on the figure shows a set of recordings, obtained for the same task, target and trial. Denote the epoch marks from one such set as $\{ep_i\}_{i=1,\dots,N_{ep}}$ where the number of epochs N_{ep} depends on the considered task and, as already stated, $N_{ep} \in \{4, 6, 8\}$, and the spike rates for the N recorded cells as $\{ST_i^c\}_{i,c=1,\dots,N}$. The spike rate for the epoch i and the cell c is calculated using the following expression

$$SR_i^c = \frac{\#\{j \mid ep_i \leq ST_j^c < ep_{i+1}\}}{ep_{i+1} - ep_i}. \quad (3.1)$$

The total number of spikes within the considered epoch is divided with the epoch length to obtain the average activity in that epoch. All the rates calculated for the same epoch i are represented by the rate vector

$$\mathbf{RV}_i = [SR_i^1 \ SR_i^2 \ \dots \ SR_i^N]^T.$$

We will denote the total collection of such vectors as

$$\mathcal{RV}(ts, tr, tg) = \{\mathbf{RV}_i\}_{i=1,\dots,N_{ep}}$$

emphasizing that it is obtained for one trial (tr) of one task (ts), for the selected target position (tg). Then, the whole database represented as the set of rate vectors is given with

$$\mathcal{D}_{RV} = \{\mathcal{RV}(ts, tr, tg)\}_{ts=1..6, tr=1..4, tg=1..8} \cup \{\mathcal{RV}(ts, tr, tg)\}_{ts=7, tr=3, tg=1..16}.$$

Assigning labels to these vectors is straightforward, the label corresponds to the epoch classification given on the Figure 2.8.

Training and test set: Taking into account all trials for all the experiments we obtain 544 data examples for the motor intention class, and 368 examples for the no motor intention one. It is important to point out the significant difference in the number of examples in the two classes, which results in an unbalanced classification problem. Straightforward implementation of the algorithm often resulted in a very biased classifier, that tended to make much more errors on the smaller class than on the bigger one.

Standard procedures for avoiding this problem, appeared to be not enough successful for our data set. Solution is found in systematical removing some motor intention examples in a four step procedure. In the i -th step, all the motor intention examples corresponding to the i -th trial are removed, and the classification is performed using the remaining data. This reduces the number of motor intention examples to 388, which is comparable to 368 of the opposite class. The procedure is repeated four times, once for each experimental trial.

The set of 756 rate vectors is used in the training and testing phase. A randomly chosen subset of the training data is used for validation. The training and validation data are utilized for selecting the optimal parameters of the classifier, while the test set serves to evaluate its performance on a 'new' data set, not used during training.

Normalization: All the rate vectors are normalized before the classification. For each cell, the spike rates are normalized to zero mean and unit standard deviation, which removes any persistent activity existing in all the experiments, and emphasizes the difference in rate modulation evoked in different epochs. Also, it is a recommendable step in the preprocessing for improving the classifier performance. As mentioned in [63], in order to ensure a fair classifier testing, the training and test sets are not normalized together. The joint training and validation data are normalized first, and the obtained parameters, the mean and the variance, are used to normalize the test data.

Classification: Varying the learning algorithms and their parameters on one side, and testing the role of some of the epochs on the other side, we selected the optimal setup for further studies. Then, the outcome was employed to test the precision for identifying the motor intention from epochs. The entire procedure is divided into four steps, as already described. In each step, motor intention related spike rates from one trial are removed, and the rest is used for training and testing. The 10-fold cross-validation is employed in each of the four steps. In other words, the data set is divided into ten subsets. Each of them is used as a test set once, while the remaining 90% of the data served for the training phase. Therefore, ten classification error estimations are obtained in each algorithm step, which gives 40 of them for all of the steps.

Results: The forty classification error estimations are used to calculate the mean error and confidence intervals according to the formulas proposed in [56] and [57]. The mean, denote it as μ , is calculated as the ratio between the total number of errors obtained in 4x10 cross-validations, and the total number of rate vectors, N_{RV} in the test set, i.e. 4 times 10% of the total number of rate vectors in the database. Three equations for evaluating 95% confidence intervals are proposed in these papers, developed under assumption of binomial or

its approximate Gaussian distribution of errors. The equations from [57] are repeated here. The textbook limits (from [56] and [57]) evaluated using the Gaussian distribution assumption are

$$\mu \pm \left(\frac{1}{2 \cdot N_{RV}} + z \cdot \sqrt{\frac{\mu(1-\mu)}{N_{RV}}} \right). \quad (3.2)$$

where $z = \Phi^{-1} \left(1 - \frac{\alpha}{2} \right)$ for the $(1 - \alpha)$ confidence interval. An alternative expression using the normal approximation developed in that report gives

$$\left(\mu + \frac{k_1}{2N_{RV}} \right) \pm z \cdot \sqrt{\frac{\mu(1-\mu)}{N_{RV}} + k_2} \quad (3.3)$$

where $k_1 = \frac{(1-2\mu)z^2}{1+z^2/N_{RV}}$ and $k_2 = \left(\frac{z}{2(N_{RV}+z^2)} \right)^2 \cdot \left[1 - 4\mu(1-\mu) \left(2 + \frac{z^2}{N_{RV}} \right) \right]$, and z is defined as before. Finally, a more precise formula based on the assumption of binary distribution of errors is given as,

$$\left(\mu + (1-2\mu) \cdot \frac{z\sqrt{0.5}}{N_{RV}+3} \right) \pm z \cdot \sqrt{\frac{\mu(1-\mu)}{N_{RV}+2.5}}. \quad (3.4)$$

In our case, all three expressions gave very similar confidence intervals and mean values, showing that the textbook formula can be safely used here.

In addition to evaluating the mean classification error, the influence of each experimental epochs is assessed. The error-per-epoch is evaluated for each of the files, and presented alongside the mean errors for the entire files. This way it is possible to compare the epochs with respect to their contribution to the total error.

Single cell influence

The importance of selecting the optimal subset from the set of randomly picked recorded cells is already emphasized in several reports. Researchers propose using only cells exhibiting high directional tuning for movement direction estimation [65, 71]. Also, studies investigating the minimal population size sufficient for operating a BCI are presented in [77, 2]. There, the neuron-dropping procedure described in [100] is applied on the entire set of recorded cells, without any preliminary selection based on directional tuning or some other desirable property.

Working with small populations in this study, enables testing the contribution of each cell in the set. This approach should reveal the characteristics of a typical randomly chosen sample of cells, in terms of motor intention coding.

Selected cells: Single or multiple cells are removed from the set before performing the described classification procedure. The result is compared to the one obtained for the complete set, using a statistical hypothesis testing. First, only one cell is removed at time and the remaining data set is analyzed the same way as before.

In the second series of tests, several cells are removed at a time. Instead of choosing a subset to be removed randomly, or trying all possible combinations, we used a simple procedure for selecting the most informative cells, described in [63]. A measure of 'similarity' between the data corresponding to the opposite classes, the bi-serial correlation, is evaluated for each of the cells. It gives an indication of the separation between two groups of spike rates, those calculated from the motor intention epoch, and those calculated for the no motor

intention ones. Denote the number of rate vectors in one file as before, N_{RV} , where N_{RV}^{mi} among them encode motor intention, and N_{RV}^{nmi} encode its absence. Also, the spike rates evaluated for the cell c in the epoch i are marked as before SR_i^c . Those obtained from the motor intention related epochs are given as $SR^{c,mi} = \{SR_i^c \mid i: \text{motor intention epoch}\}$, and the no motor intention set is represented as $SR^{c,nmi} = \{SR_i^c \mid i: \text{no motor intention epoch}\}$. Then, the bi-serial correlation for the cell c is evaluated as:

$$C_{bs}^c = \frac{\sqrt{N_{RV}^{mi} \cdot N_{RV}^{nmi}}}{N_{RV}} \cdot \frac{\text{mean}(SR^{c,mi}) - \text{mean}(SR^{c,nmi})}{\text{std}(SR^{c,mi} \cup SR^{c,nmi})} \quad (3.5)$$

This method is usually applied in problems where data vectors depend on many features, and, therefore, have big dimensionality not suitable for machine learning analysis (for example BCI interfaces based on EEG signals recorded using a big multi-electrode system). Here, we use it as a simple criteria for selecting potentially most important cells in the context of interest. Only those cells where C_{bs}^c exceeds certain threshold remain in the set. By changing the threshold value, we can control the number of removed 'least important' cells.

Finally, in some files, the obtained results left some ambiguity about the influence of certain cells. Classification using only a group of manually selected cells is performed for those cases.

Statistical testing: Each test involving cell removal results in the 40 estimations of the classification error. These samples are clearly not independent, and we have to use some of the statistical methods for comparing dependent samples. Wilcoxon matched-pairs signed-ranks test, and paired t-test for dependent samples are used, according to the procedure given in [85]. The significance level considered for both statistical methods is $\alpha = 0.05$.

In order to ensure a fair comparison between the results obtained for the same file after removing different cells, data sampling into learning, test and validation set cannot be entirely random. Instead, the random assigning of rate vectors to the training, validation and test sets according to the 10-fold cross-validation scheme is done only once. The same 10 arrangements are kept throughout the complete analysis of the same file, i.e. for testing on the entire set of cells, and for the classifications done after removing any required group of cells. This way, the error estimations in the test phase are always obtained from the same spike rate vectors, and the results are comparable.

Sliding window classification

The second approach disassociates the study from the notion of experiment epochs, and focuses on motor intention detection in time, along the spike trains. This can be seen as a more complex problem, since some spike rate vectors may be affected by processes belonging to multiple epochs. We still rely on the knowledge of epochs when composing the algorithm; but once it is completed it allows testing for the presence/absence of motor intention regardless of the epoch dynamics.

Spike rates: The entire spike trains, from the beginning to the end of a trial, rather than just a set of selected epochs, were considered. The number of data examples, as well as the temporal resolution, are defined by introducing a time window. The window is moving along the spike train in discrete time steps, with a predefined time shift. For each distinct window position, spike rates are calculated in the standard way, as the number of spikes within the

window divided by the window size. Denote the window size as W and the time shift in each step as S , then, the rate for the cells c , calculated in the t^{th} step, is

$$SR_t^c = \frac{\#\{j \mid t \cdot S \leq ST_j^c < t \cdot S + W\}}{W}. \quad (3.6)$$

and the vector of spike rates obtained from the set of N cells is $\mathbf{RV}_t = [SR_t^1 \ SR_t^2 \ \dots \ SR_t^N]$.

Window: We tested several choices for the window length, from 100ms to 1sec. These values correspond to the duration of experimental epochs illustrated on the Figure 2.9. The smallest among them (RT and MT) are several hundreds of milliseconds long, while the longest ones (Memory, for example) last for a couple of seconds. The window shift was fixed to half the window length.

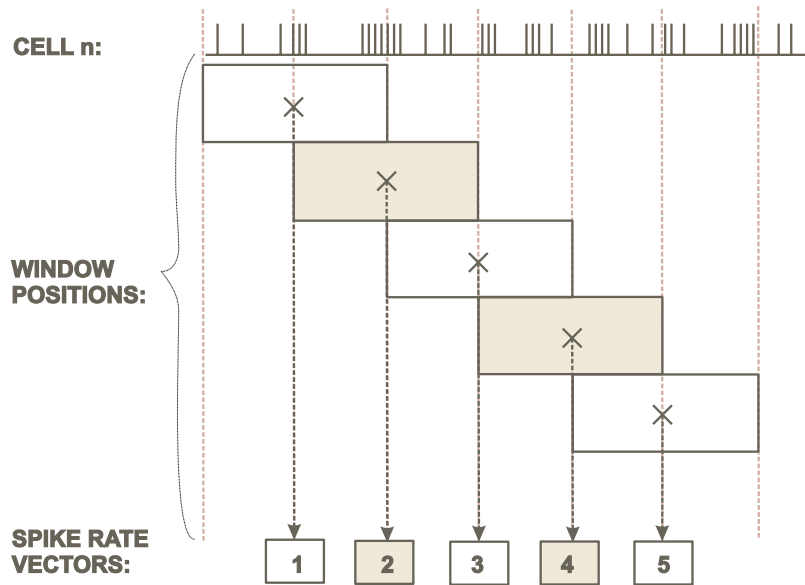


Figure 3.4: The window positions used for calculating spike rates in the second data analysis method. The fixed size window is shifted for a fixed interval of time between the two succeeding steps. The first five steps in the procedure are shown. In each step, a spike rate is calculated as an average neural activity within the window. The rates are indicated with enumerated boxes at the bottom of the figure.

The exact window shifting in successive steps is shown on the Figure 3.4. Rectangles mark window positions that are spanning one spike train (actually, a set of simultaneously recorded spike trains that have the same duration). The altering color of the windows is used only to make their positions more visible. Each window position gives one spike rates vector used for classification. For illustration, those obtained in the first five steps are marked with the boxes on the bottom of the figure enumerated from 1 to 5. The window positions are also indicated on the Figure 3.3, but without the overlapping shown, in order to keep it simpler and indicate only the major steps in the algorithm.

Classes of spike rate vectors: The calculated spike rate vectors are divided into three groups - motor intention, no motor intention, and those examples that cannot be reliably assigned to any of the two classes, as shown on the Figure 3.3. As already mentioned, the labels for spike rate vectors are determined according to the experimental epochs. If the window contains only segments of motor intention epochs the resulting rate vector is assumed

to encode motor intention. Similarly, if it contains only data from the no intention epochs the rate vector is assigned to the corresponding group. The adopted window shifting strategy ignores the a priori knowledge about epoch duration, and the spike rate vectors calculated in some steps cannot be assigned to any of the classes. For some window positions, spike rates are calculated using segments from motor intention and no motor intention epochs, at the same time. Also, for the vectors calculated using the beginning of the spike trains, where we cannot be sure about the processes present in the monkey brain, the correct labeling is ambiguous. Those data are not used for training and for evaluating the average performance, since all the examples employed for that purpose have to be reliably labeled. Once the classifier is trained, we can examine new data without knowing the epoch marks and the window position with respect to them.

Classification: The same classifier is applied as before; namely, the SVM with 10-fold cross-validation. The problem caused by the difference in size of the classes is present here as well, and the mismatch depends on the window size and the length of recordings, that varies for each file. Additionally, for a small window size, the resulting data set is relatively big with many redundant examples, which slows down the training phase, but does not change the performance of the resulting classifier.

Therefore, for small windows it is recommendable to remove some of the redundant examples and decrease the training data size. Also, if one of the classes becomes much bigger than the other, some of its examples have to be removed. The solution for these problems is found in the literature [70]. This approach employs an iterative procedure for removing the 'non-informative' examples far from the border between the opposite classes. In other words, it tends to remove the examples that do not influence the classifier properties and the way how it divides data into two classes. First the distance from the nearest 'enemies' is calculated for each spike rate vector in the set, i.e. the distance from its closest rate vector belonging to the opposite class. The pairs of rate vectors are sorted according to this information, and the pair with the biggest distance has the biggest chance to be removed. Then, the k nearest neighbors are found for each rate vector. If all of them belong to the same class as the examined vector, that vector is considered to be redundant and removed from the set. Therefore, only the spike rate vectors around the border between the classes remain. Changing the number of considered nearest neighbors, the size of the remaining set of rate vectors is adapted. Using a smaller k , the chances for removing the considered vector increase. This parameter can be chosen differently for the two classes, such that the bigger of them receives smaller parameter.

In the context of our work, this method helps in reducing the overall number of computations in the training phase, by reducing the training data set. It is also used to make the classification problem balanced, by removing more examples from the bigger class. This is an important point for removing the eventual bias in classification.

Testing phase: The testing phase in this method is twofold. First, only the labeled data are selected, and divided into the training, validation and test set, as before. The same 10-fold cross-validation procedure is applied, and the set of 10 error estimations is obtained. The average precision and the confidence intervals are calculated as before, using the expressions 3.2, 3.3, 3.4.

Then, the neural activity from one trial and for one target position is selected in advance, and used as a test set. The remaining data is divided into the training and validation sets. The employed classifier is similar as before, but instead of the classification, the estimation of probability of finding motor intention is calculated, as described in [18]. The SVM software package, described in [18], includes the algorithm that calculates the probability to belong to a certain class, rather than providing a deterministic answer to the classification problem.

It is expected to obtain a high probability of motor intention in the intervals corresponding to the sequence of reaction and movement times and in the memory epochs, and much lower probability for the rest of the recordings. Once the classifier is created, the new recordings from the same recording site can be analyzed without knowing the epoch marks.

3.4 Binary Classification Using Support Vector Machines

So far in this report, the major steps in data analysis are presented. First, the definition of the question of interest in the form of binary classification is described. The adopted approach is first explained in general terms, and illustrated on the Figure 3.3 and the related section. Then, its two variations are discussed in details. The first step in each of them is the calculation of spike rate vectors, i.e. the feature vectors in the context of a classification problem. The binary classification applied for the analysis of these new data sets is described after, the approach how to identify data representing the two classes, how to divide it into the training and test set, and how to overcome some practical problems that emerged during these tests. Finally, the statistical evaluation of the obtained error estimations is described, the calculation of the average performance, the confidence intervals, and the statistical hypothesis testing for comparing the outcomes of the different tests. Finally, the remaining point to be described is the construction of the classifier itself.

The classification method applied throughout this study is the standard support vector machine (SVM) algorithm, as already mentioned. Some well known facts related to the SVM will be repeated here, for completeness of the report. Most of these information can be found in any textbook dedicated to the SVM, for example [90, 13, 25]. Following the outline used in these books, the SVM will be first presented for the simplest problem, the binary classification of the two linearly separable sets. Then, the modification needed for solving the non-separable problem will be explained. Finally, the non-linear SVM, the tool used in this study, will be explained. The SVM possesses promising properties that attracted a lot of attention in the machine learning community in the past. The theoretical analysis of its capacity (VC dimension) and generalization properties are elaborated in many reports. For example, [13] presents some main conclusions obtained in the work of V. Vapnik and others. At the same time, methods solving the technical problems following the SVM implementation were developed [18], as well as the alternative SVM formulations that simplify the implementation and avoid some of these problems [90]. Additionally, the regression problems are formulated and studied using the modification of the SVM [90].

The following section will present only the basic SVM algorithm, as it is usually described in the literature. The advanced versions of the same algorithm, and the theoretical studies will not be elaborated here. In the context of this study the SVMs are used as a tool to solve a demanding binary classification problem, and only what is needed for understating its basic functioning will be presented.

The SVM implementation was not within the scope of this study, either. The downloadable software package, described in [18], was used for training and testing the classifier. The additional code needed, for calculating spike rates, for the cross-validation, and for the analysis and representation of the results is implemented in Matlab. All the analysis steps described in the previous sections of this chapter are implemented this way, and the SVM procedures are embedded in the Matlab functions, using its mex interface, also provided in by the same software package. Some relevant implementation details will be described at the end of this section.

3.4.1 Support Vector Machines Algorithm

Most of the SVM textbooks, introduce the method starting from the simplest case, the binary classification of the two linearly separable sets. We will also adopt that approach, first introduce the basic model and latter add the modifications needed to obtain the algorithm applicable for solving practical problems.

Linearly separable problems: Denote the considered data set as $\{\mathbf{x}_i, y_i\}_i$ where \mathbf{x}_i represents the feature vector, and $y_i \in \{-1, 1\}$ is its label in a binary classification task. Suppose the data is linearly separable, i.e. it is possible to construct a hyperplane such that all the data with $y_i = -1$ lay on one side of the hyperplane, and all the data with $y_i = 1$ on the opposite side. If that hyperplane is given by the equation $\mathbf{w}^T \mathbf{x} + b = 0$, where \mathbf{x} is a point in the space and \mathbf{w} is the vector normal to the hyperplane, then, the last statement can be described as

$$\mathbf{w}^T \mathbf{x}_i + b \geq 1 \quad \text{if } y_i = 1, \quad \text{and} \quad \mathbf{w}^T \mathbf{x}_i + b \leq -1 \quad \text{if } y_i = -1.$$

The ± 1 on the right side is obtained by normalization of the three equations. Written in a more compact form it gives: $y_i \cdot (\mathbf{w}^T \mathbf{x}_i + b) \geq 1$.

If there is one hyperplane satisfying this condition, then, there is an infinite number of such hyperplanes. Introducing an additional condition, we can limit the choice of the hyperplane and insist on some desirable properties of the model. One such condition is the maximization of the margin, i.e. maximization of the sum of distances between the hyperplane and the closest data on both sides (from both classes). It can be shown that this value equals $\frac{2}{\|\mathbf{w}\|}$, so its maximization is equivalent to the minimization of the norm of the vector \mathbf{w} , or equivalently, the square of its norm. Finally, the formulation of the problem is given as

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{\|\mathbf{w}\|^2}{2} \\ \text{s.t.} \quad & y_i \cdot (\mathbf{w}^T \mathbf{x}_i + b) - 1 \geq 0, \quad \forall i. \end{aligned}$$

The Lagrangian is calculated as

$$\mathcal{L}_P = \frac{\|\mathbf{w}\|^2}{2} - \sum_i \alpha_i \cdot (y_i \cdot (\mathbf{w}^T \mathbf{x}_i + b) - 1).$$

The considered problem is clearly convex, since the Lagrangian represents a quadratic function on \mathbf{w} , and a linear function on b and α_i . The convex problems have only global optima, therefore, any solution obtained from the previous expressions is guaranteed to represent the classifier with the maximal possible margin.

Finding the partial derivatives of the Lagrangian with respect to \mathbf{w} and b , gives a set of equations, known as Karush-Kuhn-Tucker conditions.

$$\frac{\partial \mathcal{L}_P}{\partial \mathbf{w}} = 0 = \mathbf{w} - \sum_i \alpha_i \cdot y_i \cdot \mathbf{x}_i = 0 \tag{3.7}$$

$$\frac{\partial \mathcal{L}_P}{\partial b} = 0 = \sum_i \alpha_i \cdot y_i = 0 \tag{3.8}$$

$$\alpha_i \cdot (y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1) = 0, \quad \forall i \tag{3.9}$$

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1 \geq 0, \quad \forall i \tag{3.10}$$

$$\alpha_i \geq 0, \quad \forall i. \tag{3.11}$$

Inserting the first two equations in the Laplacian gives the dual problem,

$$\begin{aligned} \max_{\alpha_i} \quad & \mathcal{L}_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \cdot \mathbf{x}_i^T \mathbf{x}_j \\ \text{s. t.} \quad & \sum_i \alpha_i y_i = 0, \alpha_i \geq 0, \forall i. \end{aligned}$$

This dual formulation and the KKT are often easier to solve than the original problem, and it is a common practice to use them instead of the original problem when training the SVM classifiers.

Linear SVM and non-separable problems: Clearly, the ideally separable problems are rare in practice. Therefore, this basic formulation of SVM cannot be directly used for most of the real-life applications. A solution is found in allowing a certain error on both sides of the hyperplane, which is formulated as $y_i \cdot (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i$, with $\xi_i \geq 0$ for $\forall i$. The parameter ξ_i accounts for the error, and represents the additional free parameters to be found through the optimization. The problem becomes the trade off between the margin maximization and the error minimization, formulated as

$$\begin{aligned} \min_{\mathbf{w}, b, \xi_i} \quad & \frac{\|\mathbf{w}\|^2}{2} + C \sum_i \xi_i \\ \text{s.t.} \quad & y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \forall i \\ & \xi_i \geq 0, \forall i. \end{aligned}$$

where C represents the trade off between the margin size and the error, If C is very small the classifier will have a margin as big as possible and make more errors on the training set. When it is large, the error minimization is dominant while the obtained margin can be small. The Laplacian for this problem is

$$\mathcal{L}_P = \frac{\|\mathbf{w}\|^2}{2} - \sum_i \alpha_i \cdot (y_i \cdot (\mathbf{w}^T \mathbf{x}_i + b) - 1 + \xi_i) - \sum_i \beta_i \xi_i.$$

and the dual problem is given as

$$\begin{aligned} \max_{\alpha_i} \quad & \mathcal{L}_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{s.t.} \quad & \sum_i \alpha_i y_i = 0, 0 \leq \alpha_i \leq C, \forall i. \end{aligned}$$

Nonlinear SVM: The last step in describing the SVM is the extension of previous results to the nonlinear problem, i.e. for the situation when the data in the opposite classes cannot be separated by a mere hyperplane. The basic hypothesis behind this step is that, although the data are not linearly separable in the original space, translating them into a sufficiently high-dimensional space results in a linearly separable problem. The previous description of

the hyperplane classifier can be extended as $y_i (\mathbf{w}^T \Phi(\mathbf{x}_i) + b) \geq 1 - \xi_i$, where $\Phi(\cdot)$ represents the function translating the original to the high-dimensional space.

The optimization problem formulated for this case is

$$\begin{aligned} \min \quad & \frac{\|\mathbf{w}\|^2}{2} + C \sum_i \xi_i \\ \mathbf{w}, b, \xi_i \quad & \\ \text{s.t.} \quad & y_i \cdot (\mathbf{w}^T \Phi(\mathbf{x}_i) + b) - 1 + \xi_i \geq 0, \quad \forall i \\ & \xi_i \geq 0, \quad \forall i. \end{aligned}$$

The KKT conditions are given as

$$\mathbf{w} - \sum_i \alpha_i y_i \Phi(\mathbf{x}_i) = 0 \quad (3.12)$$

$$\sum_i \alpha_i y_i = 0 \quad (3.13)$$

$$c - \alpha_i - \nu_i = 0, \quad \forall i \quad (3.14)$$

$$\alpha_i (y_i (\mathbf{w}^T \Phi(\mathbf{x}_i) + b) - 1 + \xi_i) = 0, \quad \forall i \quad (3.15)$$

$$y_i (\mathbf{w}^T \Phi(\mathbf{x}_i) + b) - 1 + \xi_i \geq 0, \quad \alpha_i \geq 0, \quad \forall i \quad (3.16)$$

$$\nu_i \xi_i = 0, \quad \forall i \quad (3.17)$$

$$\nu_i \geq 0, \quad \xi_i \geq 0, \quad \forall i. \quad (3.18)$$

The dual problem, which Laplacian is given as $\mathcal{L}_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \cdot \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}_j)$, becomes more difficult to solve since the scalar product of the two vectors in the expression have to be computed in the high-dimensional space. Estimation of the dimensionality of the optimal space, and explicit computation of the maps is not straightforward.

The theory offers an alternative solution, where the scalar product does not have to be directly evaluated, and the 'kernel trick' is used instead. Before describing this procedure, the Mercer's condition have to be introduced.

Mercer's condition: A symmetric, continuous function $K(\mathbf{u}, \mathbf{v})$ satisfies Mercer's condition, if for any square integrable function $g(\mathbf{u})$ it holds

$$\int K(\mathbf{u}, \mathbf{v}) g(\mathbf{u}) g(\mathbf{v}) d\mathbf{u} d\mathbf{v} \geq 0.$$

For every such function there exist a Hilbert space \mathcal{H} and a map $\varphi : R^n \rightarrow \mathcal{H}$ such that $K(\mathbf{u}, \mathbf{v})$ represents a scalar product of two such maps, i.e. $K(\mathbf{u}, \mathbf{v}) = \varphi(\mathbf{u})^T \varphi(\mathbf{v})$.

Therefore, if we could find a function $K(\mathbf{x}_i, \mathbf{x}_j)$ that satisfies the Mercer's condition, we could replace $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}_j)$ in the upper formula. This change is known as the 'kernel trick', and the function $K(\cdot, \cdot)$ is called the kernel in the SVM literature. In practice it is not easy to find such functions, candidates for the kernel. The linear and the Gaussian function (also known as the radial basis function) are known to satisfy the Mercer's condition. The polynomial kernel satisfy the condition only for the positive values of the parameters.

Calculating the dual, as before, and taking into account the kernel trick we obtain the final quadratic programming problem solved to obtain the SVM.

$$\begin{aligned}
\max \quad & \mathcal{L}_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \\
\alpha_i & \\
\text{s.t.} \quad & \sum_i \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad \forall i.
\end{aligned}$$

This formulation enables calculating the set of parameters $\{\alpha_i\}_i$ that corresponds to the global maximum of the dual function. For the new feature vector \mathbf{x} , the class label is obtained as

$$y(x) = \text{sgn} \left(\sum_i \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b \right). \quad (3.19)$$

We could notice that the only feature vectors \mathbf{x}_i from the training set which contribute to the (3.19) correspond to the values $\alpha_i \neq 0$. The equations (3.14)-(3.18) permit the following two possibilities. When $\alpha_i = 0$ we have $\nu_i = C > 0$, $\xi_i = 0$. The corresponding feature vectors do not contribute to the equation (3.19), and do not cause error in classification, i.e. they are far from the margin. Otherwise, $\alpha_i > 0$ and $y_i (\mathbf{w}^T \varphi(\mathbf{x}_i) + b) = 1 - \xi_i$. These vectors, that contribute to the equation (3.19), are either on the margin or close to it, and classified with some error. These feature vectors that, practically, define the border between the two classes, are called the support vectors. The last equation for the support vectors can be used to calculate the parameter b that is not directly obtained through the optimization procedure.

Together with the theoretical derivation of the SVM classifier, two additional issues have to be taken into account. Solving the given quadratic programming problem is not a trivial task, and several techniques to combat that problem are proposed. In [18] an iterative procedure is implemented instead of solving the original quadratic programming problem directly. In each iteration, the two parameters α_i are selected, and the corresponding quadratic equation is solved for them. The old values for these parameters are replaced with the new ones, and the procedure is continued until the convergence. Another procedure for simplifying the optimization problem is proposed in [90], where the quadratic programming is replaced with the linear programming.

The implementation of the SVM training was not of interest in this study, and we have used the functions from the publicly available toolbox described in [18]. The only question related to the SVM design that was considered here was the optimal choice of the 'free' parameters. As already stated, the parameter C in the given expressions is not selected automatically, but is considered as a fixed value during the optimization. Other parameters that have to be selected before the optimization are those describing the kernel function. We used the most common type of kernels, the radial basis kernel, in this study. This function is given with the expression

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\mathbf{x}_i^T \mathbf{x}_j}{\sigma}\right). \quad (3.20)$$

that depends on only one parameter, σ , the kernel width. The tests carried in order to examine the optimal choice of the two parameters, C and σ , are described in what follows.

3.4.2 Implementation details

The SVM algorithm was presented in the previous section, but only the most important information is described, without the implementation details. All of the methods used here

are the standard textbook models and the standard publicly available software, and in the context of this work there is no need to further elaborate them. Still, some details, specific for the solution of this problem, should be described. We will pay more attention to the selection of the two free parameters of the SVM, and give some additional arguments for using this particular type of classifiers with the described cross-validation technique.

Parameters selection through validation: Various strategies for selecting the parameters C and σ are listed in the literature. The simplest and most common one is to test all the values on a 2D grid. The 'nodes' on the grid should span a large range of values, so they are chosen on a logarithmic scale. For selecting the optimal parameters a part of the training set, typically 20%, is employed for validation. The remaining training data is used to solve the original optimization problem for the fixed values of C and σ , and the performance of the obtained classifier is evaluated on the validation data. The pair of values that minimizes the error on the validation set is selected, and used latter in the test phase.

Various advanced strategies for parameters selection are also presented in the literature. For example, [88] describes a method for iterative refinement of the grid-search strategy, while maintaining a reasonable training time. First, the values on a coarse 2D grid are tested, and in each step a new set of points is chosen in the neighborhood of the most successful point from the previous step. A procedure for the automatic selection of all the SVM parameters, solving the two optimization problems in each step, is proposed in [19]. The α_i parameters are computed as before, assuming some fixed values for C and σ . Once the classifier is obtained, a function describing the validation error estimate is minimized to calculated the new values of C and σ . The gradient descent algorithm is employed in this step. The method is iterated until the minimal error estimation is obtained. Finally, in [31] a third approach, based on an evolutionary algorithm, is presented. The free parameters are represented as an m -dimensional vector, and a goodness of fit (i.e. an error measure) is defined. In each step, an offspring of $n > m$ parameters is generated by calculating the center of mass of the old m parameters (intermediate recombination) and adding a random value to each offspring (mutation). The random values are derived from the zero-mean Gaussian distribution. The goodness of fit is tested for each parameter and the best m among them are stored for the next step. The iterative procedure is stopped when it reaches some termination criteria.

For choosing the two free parameters, the kernel width and the regularization coefficient, we tested the following three approaches: the standard grid search, the simulated annealing, and the random selection of parameters from an empirically determined distribution. In the grid search, all the pairs from $(C, \sigma) \in \{2^{-5}, \dots, 2^{20}\} \times \{2^{-10}, \dots, 2^5\}$ are tested and the one that gives the minimal validation error is selected.

The simulated annealing is implemented using the publicly available Matlab software, developed according to [49, 48] and given in [97].

In the last method, the parameters were chosen from a Gaussian distribution, selected empirically. First, the grid search for the parameters

$$(C, \sigma) = \{2^{-30}, 2^{-25}, \dots, 2^{30}\} \times \{2^{-20}, 2^{-15}, \dots, 2^{20}\}.$$

is completed and the optimal parameter subspace was selected. For most of the tests, the rotated 2D Gaussian distribution approximated well the optimal subspace, and it was employed for selecting the values to test. The employed distribution is given as

$$[\log_2 C, \log_2 \sigma] \sim \text{Gauss}(\mu, \Sigma)$$

where for the first method, the epochs classification, the mean value and the covariance matrix are given as

$$\mu = [10 \ -5], \Sigma = V_s \cdot \begin{bmatrix} 2.5 & 0 \\ 0 & 1 \end{bmatrix} \cdot V_s^T \text{ and } V_s = \begin{bmatrix} 4 & -1 \\ -1 & 4 \end{bmatrix}.$$

For the second test, the sliding window classification, these parameters are chosen as

$$\mu = [10 \ -10], \Sigma = V_s \cdot \begin{bmatrix} 28 & 0 \\ 0 & 8 \end{bmatrix} \cdot V_s^T, \text{ and } V_s = \begin{bmatrix} 1 & -\frac{3}{4} \\ -\frac{3}{4} & -1 \end{bmatrix}.$$

The distribution was chosen as to include the optimal subspaces for all the analyzed data sets. As usual, instead of looking for the optimal parameters directly, we used a log-log scale. The 100 randomly chosen pairs (C, σ) are tested in the validation phase, and the best among them is selected for the test phase.

In general, none of the three methods exhibited superior performance, and for the presented study, we adopted the last approach. Simulated annealing is, theoretically, the best candidate, but here it did not improve the results; on the other hand, the algorithm training time was considerably increased. The first and the third methods were very similar in both precision and efficiency; we chose the third one because it clearly avoids testing some very unsuccessful candidates.

Cross-validation: As already described in the previous sections, the 10-fold cross-validation is adopted for estimating the classifier performance. Some other methods for inferring the statistics of the method performance were also tested. The tests with N -fold cross-validation with $N > 10$ are also employed but neither the mean nor the confidence intervals differed significantly from those obtained using the adopted method. At the end, the least computationally demanding method was used. Also, the bootstrapping for the estimation of error statistics is tested according to the receipt given in [25]. This method gave tighter confidence intervals but also exhibited a systematic bias when estimating the mean error.

Other tests: The first method tested for solving the presented problem was the linear classifier. This method was not able to solve the problem, and did not perform better than the pure chance. We repeated the same analysis with a classifier of multi-layer perceptron type, which resulted in a similar performance as the one obtained using the support vector machine. The latter was selected for the final version due to easier tractability of the values of its parameters, and generally good properties of the SVM classifiers.

Chapter 4

Experimental data study

In the previous chapters we presented the detailed description of the methods used for data analysis. The neural activity is represented by the average firing rate, following the assumption that the motor intention can be inferred from the modulation of spiking intensity. The two approaches to the rate calculation were described in the previous section. The rates obtained from a set of simultaneously recorded spike trains were used in the same decoding task. In order to decode motor intention from the data we formulated a binary classification problem, and employed a standard SVM classifier to solve it. Some basic notions explaining this type of classifiers were presented, without intention to further elaborate the theory behind it. Only the problem specific aspects of the SVM tuning are described. Therefore, the details about the implementation of the training and test functions are not included, but the tuning of the free parameters was considered.

In this chapter, we will present and discuss the results of the described analysis. First, the average classification error for each of the files in the database is presented in the Section 4.1. In the Section 4.2 these results are further elaborated by analyzing the error distribution across the experimental epochs. Another interesting issue, the contribution of single cells from the recorded population is assessed in the Section 4.3. The third considered parameter is the length of the window used to calculate the average neural activity in the tests described in the Section 3.3.1. The obtained conclusions are shown in the Section 4.4. Finally, at the end of the chapter, in the Section 4.5, an additional discussion of the obtained results is presented.

4.1 Epochs classification results

The first series of data analysis tests is carried using the method described in the Section 3.3.1. Using the definition of motor intention and knowing the key processes in each step of the execution of a movement, we related some of the experimental epochs to the motor intention. Similarly it was possible to identify the epochs where the motor intention does not exist. This hypothesis is tested through data analysis. For each of the epochs, the spike rates are calculated from the set of recorded spike trains, and these vectors of rates are used for training and testing the SVM. This set up enables testing the main assumption behind this work, precisely, is it possible to identify motor intention from the neural activity, and if so, how reliable the procedure is. It, also, represents the first phase in studying motor intention in a typical reaching movement. This framework was used to select the best data analysis method, and all the later tests employed the same algorithm. The only difference being an additional tuning of the free SVM parameters. As noted before, we attempted classification with artificial neural networks. Tests with both, linear and nonlinear multilayer perceptron networks, were performed. The linear classifier was unable to outperform the classification

by chance. The nonlinear network produced similar results as the SVM classifier, and it was less tractable than the SVM.

We, also, examined the influence of the CT epoch discarded from the final algorithm version. The same classification is repeated for the CT epoch included among the no motor intention examples. The obtained error was not significantly different than the presented one, but it was not straightforward to explain the role of this epochs. In another test we removed the last MT (MTe in the ME task, and MTh in R, RF, MR and MRF) from all the experiments, since these epochs can be considered as the realization, rather than planning of a movement. The average error rate was not significantly different than the one on the Figure 4.1. The complete results of these tests are examined again in the next section, when discussing the classification error per epoch.

As already stated, each file, i.e each set of recordings with fixed electrodes position, is analyzed separately. One classifier is trained and tested each time and one set of 40 classification error estimations is obtained in the test phase. The percentage of misclassification is evaluated according to the expressions (3.2)-(3.4). The confidence intervals are calculated according to the same formula.

The summary of the results obtained using this method is presented on the Figure 4.1. The two panels on the figure correspond to the two available databases, recorded from the two lab monkeys. The classification error per file is presented alongside the number of recorded cells, the x axis shows the number of cells, while the y axis gives the error rate. Each point on the figure shows the mean error and the confidence interval obtained for one file. The results are sorted according to the increasing errors, in order to show the general trend in the classifiers performance. The tests on the second database outperform those on the first one; the precision obtained for the majority of files falls between 25% and 30% of misclassification, as opposed to 30% to 35% for the first database. The best and the worst results in the first database are around 25% and 40%, while for the second database they are 18% and 32.5%. Even in the worst case, the results of both databases outperform classification by chance, that would yield around 50% error.

As already mentioned, the x axis shows the number of recorded cells for each file. Evidently, it is difficult to observe any relation between the recorded set size and the obtained error. The recording procedure randomly picks cells, with the strength of the observed activity as the only criteria for the electrode placement. The information encoded in the obtained activity, and the cell involvement in the particular tasks cannot be assessed at that stage. Therefore, it is expectable that some of the collected cells do not contribute to motor intention coding, and their presence in the set does not significantly influence the algorithm performance. Also, it is possible that some of the electrodes pick the cells of correlated behavior, so that some of them encode redundant information, already represented in the activity of other cells in the set. This issue will be analyzed in details in the Section 4.3

The performance illustration on the Figure 4.1, describes the average ability of the proposed method to identify motor intention from the data. Also, it examines the limits in its performance. Additionally, the results of the most illustrative files shown on the Figure 3.1 should be emphasized. The average errors obtained for them are 26.14% for 'Cru_33', 30.94% for 'Cru_29', 28.44% for 'Uma_7', and 22.89% for 'Uma_11'. Considering only these four files, it is difficult to draw any general conclusion regarding the influence of the particular epochs and cells on the algorithm performance. The file 'Cru_33' exhibits a smaller classification error than 'Cru_29' in spite of the smaller number of recorded cells, at the other hand, the two files from the database 'Uma' contain the same number of cells but exhibit a difference in errors of around 5%. The file 'Cru_33' might indicate that the presence of cells whose activity can discriminate between the NGO and VIF on one side, and all the other rate vectors on the other side improves the performance. Still, for both examples from 'Uma'

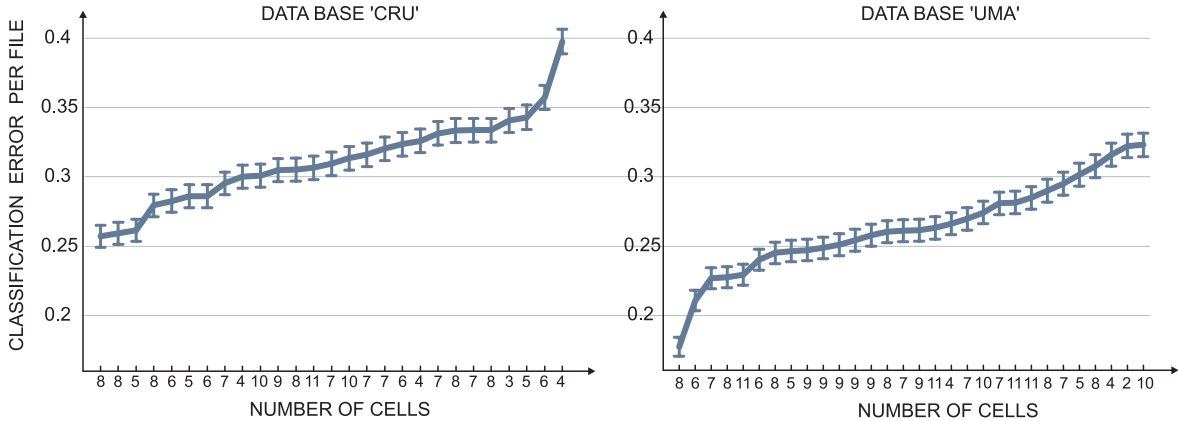


Figure 4.1: Error per file for the epochs classification. Panel on the left - database 'Cru'; panel on the right - database 'Uma'. The x axis shows the number of recorded cells in each file. The y axis shows the average classification error for each file, and the corresponding 95% confidence intervals. The results are sorted with respect to the mean error, in order to show the minimal, maximal, and average performance of the method.

(and particularly in 'Uma_7') the relation between the NGO and VFIX is not visible, but the obtained errors are similar to the one for 'Cru_33'.

The results presented in this section, the average classification precision per file, show that the implemented method can achieve between 70% and 75% correctly classified examples in an average file, and 82% in the best case. This is significantly better than the precision that would be obtained when randomly assigning data to the classes (50%). Still, we cannot infer the reasons for having such results from these tests. The recorded neural activity significantly varies in different experimental epochs, which is very expectable according to the neurophysiologic literature, and also visible on the Figure 3.1. In fact, most of the neurophysiologic studies of functional properties of the parietal cortex (as well as other regions) examine the spike rate modulation obtained under various experimental paradigms, for example different target positions, or tasks, or epochs within tasks. This modulation is sufficiently variable in different epochs to enable drawing conclusions about the cortex region of interest. This also holds for the previous work carried on this same database, presented in [9, 10]. Therefore, it is necessary to further explore these differences and their influence on the classification precision.

4.2 The error distribution over experimental epochs

The results presented in the previous section show the average performance of the applied method and reveal how reliable is motor intention decoding from the data. Still, this representation, averaged across all the experimental epochs, gives no information about the contribution of each epoch. The diversity of the obtained neural activity suggests significantly different influence of different epochs. Therefore, it is interesting to closer investigate the classification error distribution across the experimental epochs, and the results are presented on the two panels of the Figure 4.2.

The errors for each of the epochs in each folder are presented in a color coded matrix, with the dark blue corresponding to zero-error, and dark red to the maximal misclassification (more than 90% of the examples are placed in the wrong class). The precise relation between the colors in the matrix and errors is shown with the color bars placed left most on both of

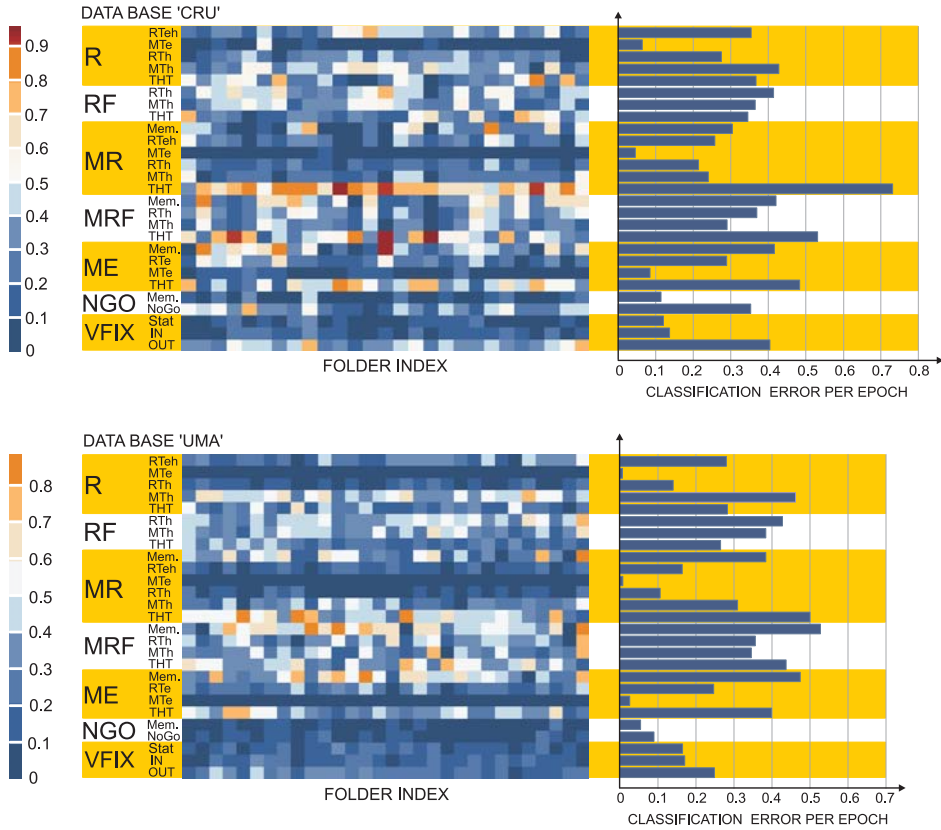


Figure 4.2: Classification error for each of the experimental epochs. Upper panel - database 'Cru'; lower panel - database 'Uma'. The matrix of errors for each epoch in each file is color coded; dark blue - zero error, dark red - the maximal misclassification. The left most bar on each panel shows the error rate corresponding to each of the colors. Tasks and epochs list shown relative to the error rate matrix. Right most panels - the average error for each epoch, average over all the files.

the panels. At the beginning of each row in the matrix the corresponding epoch and task are denoted. The bar plots on the right show the average error per epoch, averaged over all the files in the database, in order to emphasize the difference in classification for different epochs.

Several conclusions can be drawn from this figure. Clearly, NGO and VFIX examples exhibit less errors than the other tasks. In the remaining five tasks, the THT epoch significantly increases the total error. The Figure 3.1 indicates such a result, since spike rate vectors calculated from THT epochs of both classes often share similar properties. Among the first five tasks, RF and MRF, i.e. the tasks where the animal keeps the eyes fixed to the center of the screen and reaches with the hand only, show worse result than R, MR and ME. This might indicate that the intention for eye movements have more impact on the neural activity in the considered region than the intention for hand movements. Additionally, in the R, MR and ME, the best results are obtained during the MTe epoch, i.e. epoch related to the eye movements and the preparation for hand movements, which might support this statement. Still, the activity in this most successful epoch is modulated by the early planning of a hand movement, together with the actual eye movement. It is not clear which of the two signals have bigger impact on the classification precision. The region 7 is considered to be a visual area in the neurophysiologic literature, i.e. its activity is dominantly related to the eye movements. Still, the previous studies on this database, [9, 10], show the existence of eye dominant

cells, hand dominant cells, together with the combinatorial cells influenced by both effectors.

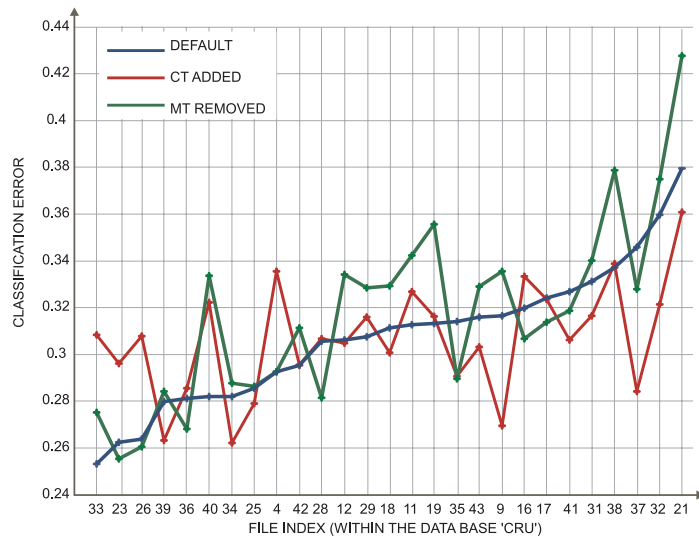


Figure 4.3: Classification error per file for the database 'Cru'. The comparison between the result presented in the previous chapter, the test with included CT and the test with the excluded MT epochs. The x axis shows the file index, and the y axis classification errors.

As already mentioned, two additional tests with somewhat different arrangement of epochs were carried out in order to obtain more insight into the considered problem. The same classification procedure is repeated here, and the only difference is in the choice of considered epochs. The results are summarized on the Figure 4.3 and Figure 4.4, with the same notation used as in the Figure 4.1 and Figure 4.2, respectively.

In the first additional test the CT is included. This epoch was not considered in the previous tests since it was not clear which processes may influence the recorded activity. The only command given before this epoch is the instruction to fixate the center and wait. Still, it is not excluded that the monkey already plans a movement anticipating the incoming signal. In this test we assumed that CT does not encode motor intention. Adding this epoch, the number of examples in the two classes becomes 544 and 560, which is sufficiently close. Therefore, the procedure with removing data from one trial was not needed, and only the standard 10-fold cross-validation was performed.

The second test aims to examine the influence of the last movement time in each of the reaching tasks. The epoch MTh in the R, RF, MR and MRF was removed, together with the MTe from the ME task. As already stated, these epochs might be related to the movement execution instead of its planning, since all the movement preparations are already completed here. Therefore, some of the examples assumed to encode motor intention are removed in this test. The number of examples in the two classes is 384 and 368, which is, again, sufficiently close to allow using the simple 10-fold cross validation without additional adaptations.

The Figure 4.3 compares the results from the test presented in the previous section, with the two additional tests explained here. All the results are shown for one of the databases, since similar conclusions hold for both of them. The blue curve is the result already presented on the left panel of the Figure 4.1, and is repeated here as the reference. The red curve illustrates the outcomes of the test with the included CT epoch, and the green one corresponds to the test with the removed MT epochs. The x axis shows the index of the files in the database, and the y axis the classification error, as before. On average, none of the three methods visibly outperforms the others. Just for some files one of them can be seen as superior, but this is

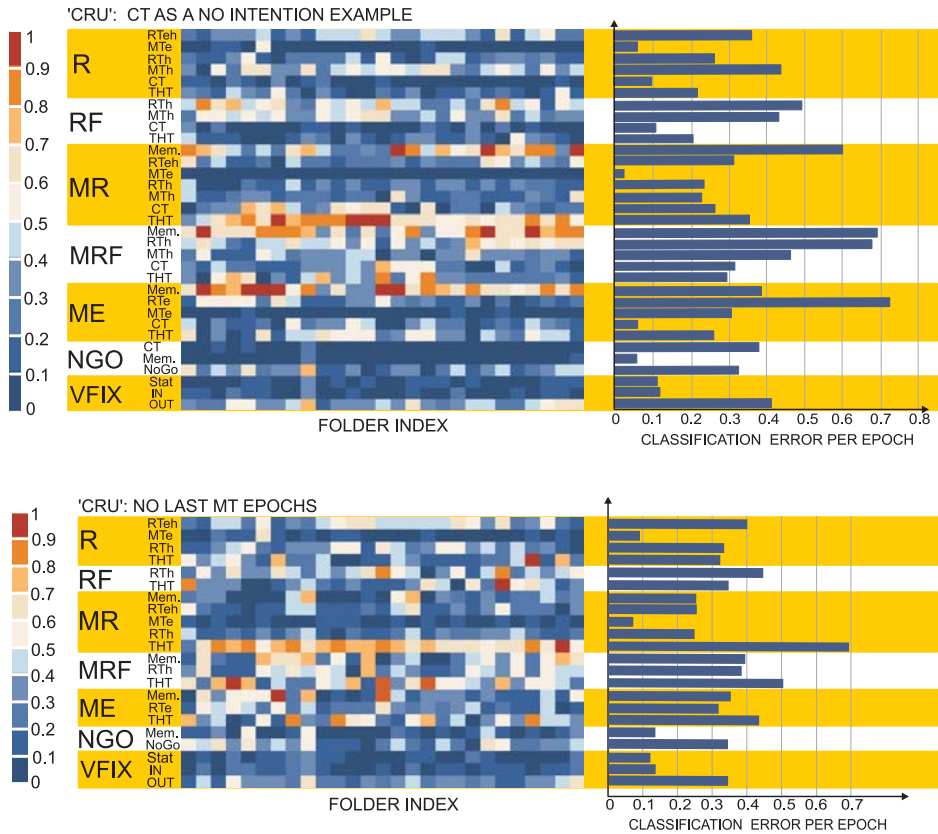


Figure 4.4: Additional tests with different epochs arrangement. The same notation as in the previous figure. Upper panel - the CT epoch is included as the example of the absence of motor intention. Lower panel - the last MT epoch is removed from the set in order to test weather it contribute to motor intention or to the motor execution. Both additional tests show the similar performance as the one presented in the Section 4.1. The difference in distribution of errors across epochs can be observed.

not systematic.

On contrary, the Figure 4.4 reveals certain differences. The test where CT is included as the no motor intention epoch (upper panel on this figure) is compared with the test where CT is not included at all (Figure 4.2). It is evident that the NGO and VFIX results do not change. On contrary, for the remaining five tasks, the errors on the RT, MT and Memory epochs increase, while the classification error for THT becomes smaller, as a result of including CT. The CT itself gives relatively small classification error compared to other epochs. One possible explanation for this is that CT have similar properties as the THT epoch, at least for the same task. Therefore, the number of spike rate vectors similar to THT becomes, effectively, two times bigger in the training set. The obtained classifier is able to represent better the THT and CT, and the error on the test set becomes smaller. Still, this has a negative impact on the classification of the other epochs in the same tasks, due to the similarity of examples from the opposite classes calculated from the same task.

The results obtained from the second test are more similar to those from the Figure 4.2. All the remaining epoch on the lower panel of the Figure 4.4 perform the same as before, and the only difference is in removing the MT epochs. The MTh in the tasks RF and MRF significantly increases the total number of errors, but the MTe in ME is one of the epochs with the smallest percentage of misclassification. Therefore, their influences compensate in the

referent test, and the two compared results are not significantly different, as already stated.

4.3 Single cell influence in a small population

As already described in the Section 3.2.3, the optimal selection of cells used to operate a BCI is an important issue. Some authors propose using only well tuned cells [65], while the others aim to estimate the minimal number of randomly selected cells needed for reliable control of a BCI [100]. In [76] the danger of overfitting and the computational burden of an algorithm were taken into account when proposing the optimal method to select cells. Here, due to the small number of recorded cells considered simultaneously we adopted a different approach to investigate the cell influence. We used the 'leave one cell out' strategy to test the influence of each cell on the performance of the entire population. In each step, one cell from the set is removed and the same classification as before is carried using the remaining data. It is repeated for each of the cells. Some additional cell removing strategies were also tested. The method that removes only the least selective cells based on the similarity of their spike rates observed for the two classes is implemented, as described in the Section 3.3.1. These tests were not always sufficient to explain the properties of each cell. Therefore, some additional tests on manually selected cells were needed to resolve the remaining ambiguity.

Some illustrative examples of single cell influence to the classification precision are given on the Figure 4.5. Two columns on the figure present the selected examples from the two databases. Each panel shows the comparison between results obtained for one or several removed cells (gray line) with the one obtained for the entire set of cells, the reference value (red line). The results that are significantly different from the reference result (compared using Wilcoxon and paired t test for $\alpha = 0.05$, as described in the Section 3.3.1) are marked on the figure. In total, for 67.4% of all the cells in the first database, and 75.1% in the second one no change in the classification error is observed when the cell is removed from the set. On contrary, 28.8% and 20.6% of cells significantly change the result. For the remaining cells, 3.8% in the first, and 4.3% in the second database, it was not possible to reliably determine this information, since the two mentioned statistical tests gave different results.

The two figures in the upper row (files 'Cru_36' and 'Uma_11'), are the most usual outcome from this analysis. Some of the cells in the set have a visible impact on the classification accuracy, since removing them leads to a significantly worse result. Still, for the majority of cells excluding them from the set does not lead to a statistically significant change of the classification error. Removing more of them, on contrary, produces the visible error increase. For the file 'Cru_36', removing all the cells that do not impact the result when considered one-by-one, increases the error showing that this subset of cells carries a relevant information as well. In the second example, 'Uma_11', removing the two 'least important cells' does not change the result, but repeating that for the three cells shows a significantly bigger error .

The two panels in the middle (files 'Cru_33' and 'Uma_32'), show the sets that contain some cells that influence the result, and the others that can be removed from the set without consequences. Clearly when repeating the analysis with cells previously marked as important, the result does not change with respect to the one obtained for the entire set. The example 'Cru_33' is also illustrated on the Figure 3.1, which emphasizes the role of the cells 5 and 9, while the analysis shows that cell 12 have equal importance, although it is not obvious by looking at the data.

Finally, the last two examples show cells that degrade the classification performance when present in a set of recordings and their removal improves the classification error, a situation that was rarely occurring. There is one such cell identified in the panel on the left. The example on the right is somewhat more interesting. Single cell removing procedure shows

two cells that have a significant impact on the result, and the remaining five to be less important. When repeating classification with only the first two cells, the result is improved, which indicates that the remaining five cells, considered together, carry an information that degrade the result. This result might seem contradictory. Still, having in mind that multiple information are encoded simultaneously in the cortex activity, it can be assumed that some of the recorded cells become involved in the representation of a completely unrelated information and do not contribute to motor intention coding.

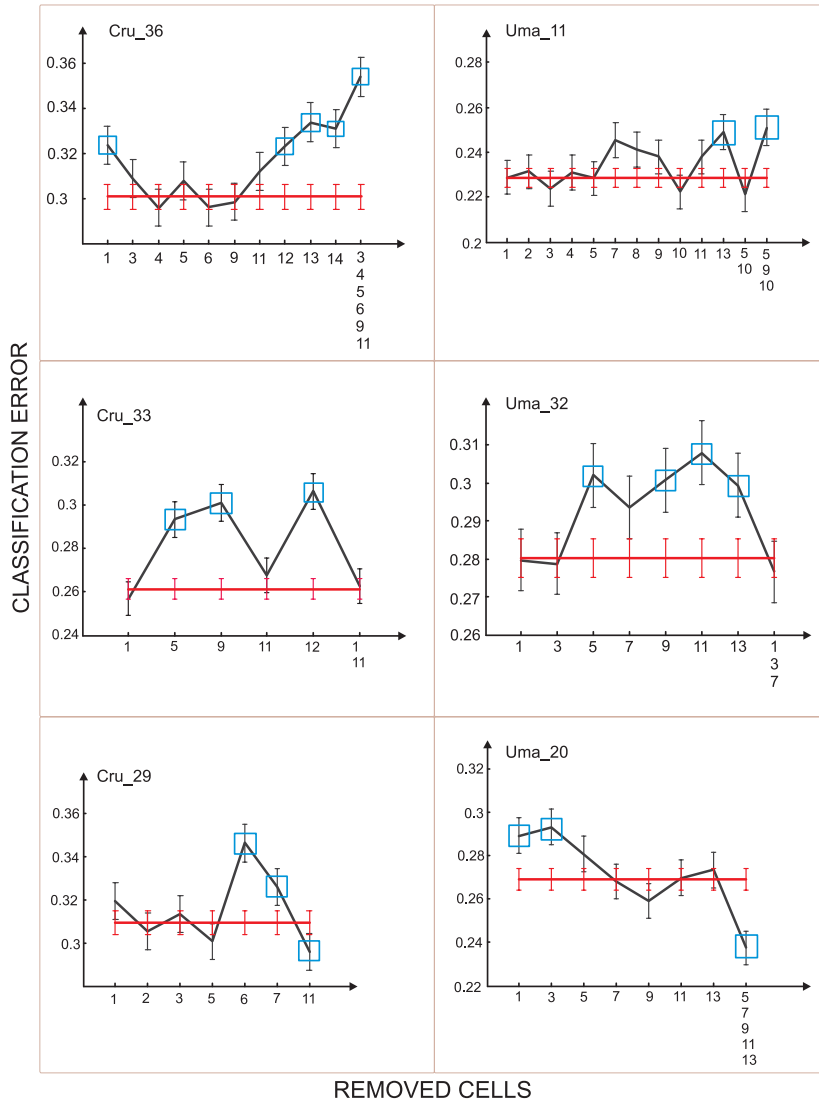


Figure 4.5: Tests of the influence of single cells. Each of the panels present the result for one file; left panels - database 'Cru', right - database 'Uma'. The indexes of the removed cells are shown on the x axis, the classification error is presented on the y axis. Red line - the reference classification error obtained using all the available cells in one file. Grey line - classification errors obtained after removing one or multiple cells from the set, and repeating the standard analysis for the remaining data. The values that are significantly different (Wilcoxon and paired t test for $\alpha = 0.05$) from the referent value are marked with boxes.

The last observation indicates that some cells can harm the performance, but they are rarely encountered. Instead, most of the cells either contribute to the result, or, at least, do not influence it. Since we are restrained to small populations, which minimizes the possibility

for overfitting, it is recommendable to use all the available cells. A single cell cannot achieve sufficient precision of classification, even for the optimal type of cells, selected according to some criteria. The information of interest is encoded in a distributed way in the cortex, and at least a small set of cells is needed for a certain reliability of classification. Increasing the population size is likely to improve the algorithm performance. This assumption will be later tested on an artificial model.

It is important to emphasize the dominant representation of cells that do not influence the classification performance alone, but in a combination with other cells. This might indicate that the same information is encoded in a redundant way in the activity of these cells, and that removing one or several of them does not remove the relevant signal.

4.4 Sliding window classification results

The last parameter of interest that we tested within this study is the integration time used for calculating spike rates. The algorithm for calculating rate vectors is modified with respect to the one discussed so far. Previously, one single rate vector was evaluated for each epoch in each task, and the integration time varied a lot for different epochs. Some of them, like RT and MT, lasted even less than 100 milliseconds, and often between 100 and 200 milliseconds. On the other hand, the epochs like Memory and THT were even a couple of seconds long, as illustrated on the Figure 2.9. Therefore, it is not possible to assess the influence of the integration time in this context. Instead, an alternative method for calculating spike rates is introduced, a sliding window that segments the recordings for rates evaluation. The window spans the entire length of recordings, moving for a fixed interval in each step, as described on the Figure 3.4. This fixed shift is set to half of the window length. Varying the length we could assess its influence on the classification performance. The drawback is that assigning labels to the rate vectors becomes more difficult, since it requires tracking the window position with respect to the epochs. For those windows which cover one or several epochs of the same type, i.e. motor intention or no motor intention epochs, the label is chosen in accordance with those epochs. On contrary, if the window crosses a boarder between the epoch belonging to the opposite classes, it is no longer possible to relate it to one class. We attempted to examine the probability of finding motor intention in such examples, instead of doing a simple binary classification.

This method can be seen as the ultimate goal of one such analysis. It aims to identify the presence of motor intention wherever it appears in a recording. The method is no longer dependent on particular tasks and epochs. The epoch marks are needed in the training phase, but for the new tested examples this information is not needed. No particular limitations of the type of movements or the number of targets is imposed, the movement can be executed to one from the discrete set of targets, or to any random position in the space. The only important issue is to collect the training set that sufficiently well represents the selected experimental setup.

The average performance for this method is illustrated on the Figure 4.6. Left and right columns correspond to the two databases, as before. The upper two figures show classification error evaluated for different windows size (300msec, 500msec, 700msec and 900msec). The blue curve corresponds to the 300 milliseconds window, the green one to the 500 milliseconds, gray to the 700 milliseconds, and red one to the biggest window tested of 900 milliseconds. The x-axis gives the number of cells in each file, and the classification error is given on the y-axis. The mean values and confidence intervals, based on the 10-fold cross-validation, are calculated as before. The general tendency suggests that the classification error increases as

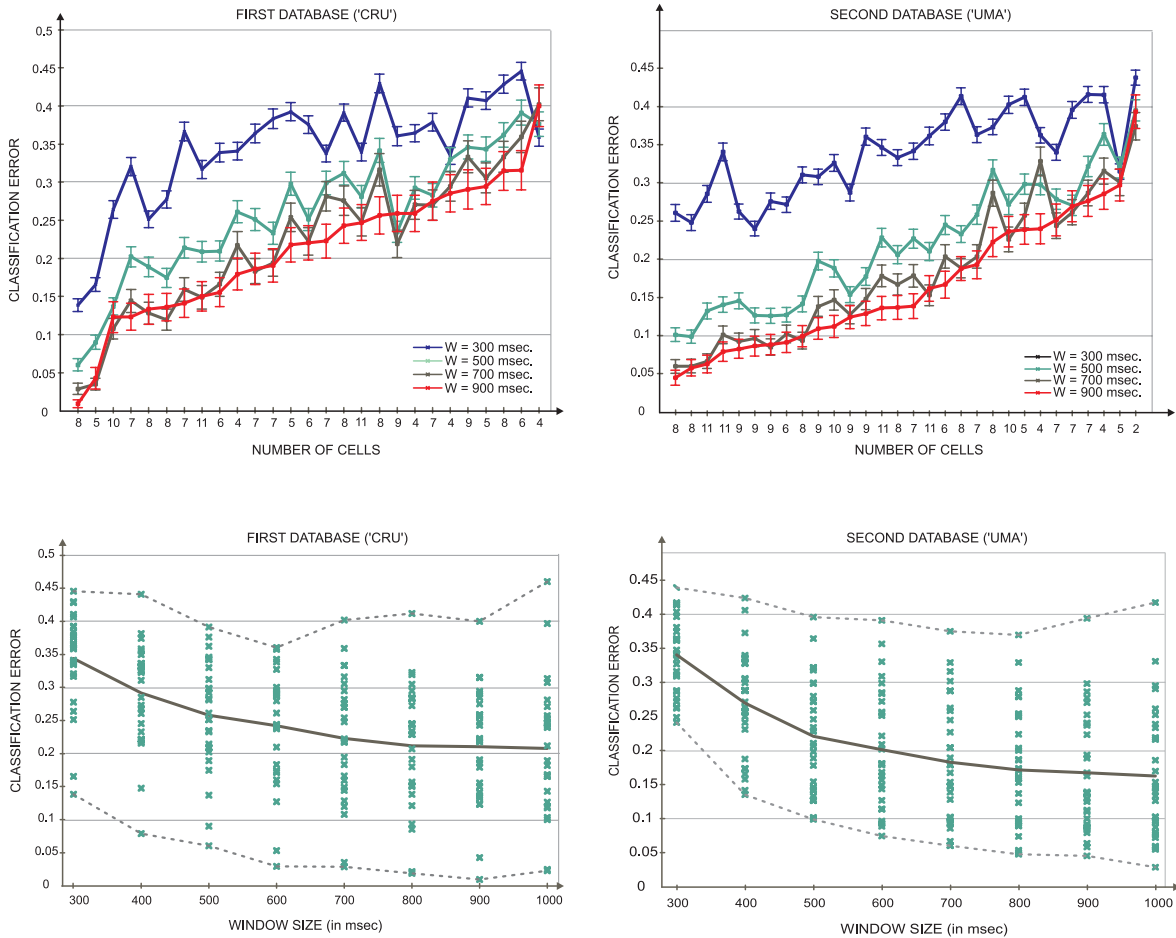


Figure 4.6: Summary of the results obtained from classification for the fixed window size. Left column corresponds to the database 'Cru', right to 'Uma'. Upper panels - dependency of the error rate on the number of cells and the window size. The number of cells available in each file is shown on the x axis, the y axis gives the percentage of misclassification in the test set. The results obtained for the four different windows are shown - red, gray, green and blue correspond to $W = 900, 700, 500, 300$ milliseconds, respectively. Lower panels give an alternative representation of the same results. The confidence intervals are not shown here, and each green cross corresponds to the average error for one file and one window size. The results are arranged with respect to the window size in order to emphasize the major trends, i.e. the error decrease with the increase of the window size. Additionally, the large variation of the results obtained for the same window size is illustrated.

the window length decreases, which is in accordance with the results reported in the literature [77].

The classification precision varies for different files even more than in the previous tests. It goes from around 5% error for the best examples, to 40% error for the worst ones. This is even more pronounced on the lower two panels, where only the mean values of classification errors are reported for each file at different window sizes. Each green cross on the figure gives the mean classification error for one file for one selected window size. The results for a fixed window size are shown in one column without indicating any additional information about the corresponding files. Such visualization is used only to show the variation in results and the mean error, how it decreases as the window increases. This is illustrated with the middle

gray line, the average calculated over all the errors for single files.

For a small ad hoc chosen set of cells, the results show good discrimination between presence and absence of motor intention, but if the chosen set is not so 'informative' the algorithm performance rapidly decreases. In general, the longer is the window used in the analysis, the better is the obtained result. Here, it is particularly critical since very small groups of cells are used in the analysis. The lack of information due to the small population size is compensated with the longer integration time of the information. This requires using relatively big windows compared to the epoch size and those proposed in the literature [2, 100]. Note that some of these studies use a relatively long total portion of recordings for building a model, although the integration time for calculating spike rates is small. For example in [16] the integration time is 100 milliseconds, but the last 10 intervals are used to predict the movement direction, i.e. one second of activity is used for prediction. In the context of this work, we aim to test for the presence of motor intention from each segment of the data defined by a certain window position. Evidently, a certain minimal time is needed to collect the information that is discriminative enough for a classifier. In [65], for a set of cells of comparable size as those used here the employed integration time is 900 milliseconds, which also gives good results in our study.

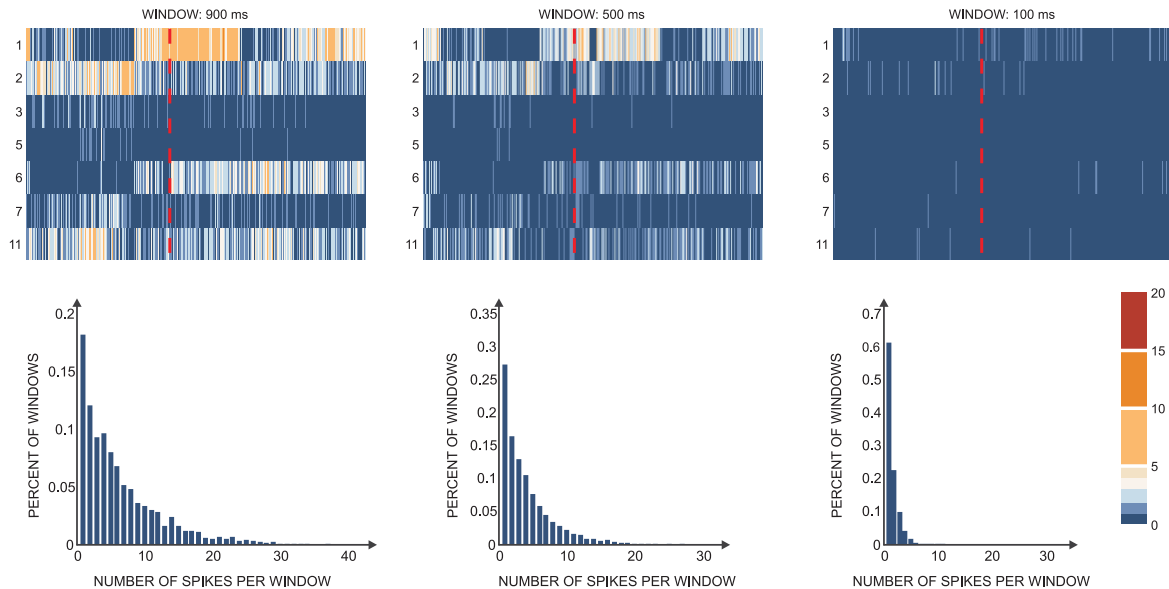


Figure 4.7: Spike count per window, for the file 'Cru_29'. Upper panels - matrices where each row corresponds to one cell, and each column to the number of spikes counted within one window. Illustrated for the windows of 900, 500 and 100 milliseconds, from left to right. Color coding of spike counts is shown on the bar, right most in the lower row. Lower panels - percentage of windows containing a certain number of spikes. x axis - the number of spikes (from 0 to 36 spikes), y axis - the percentage of windows containing certain number of spikes given on the x axis. These illustrations show the diversity in the set of spike rate vectors. For the smaller windows this diversity is less pronounced, which can be the explanation for the decrease in the classification accuracy.

Data visualization presented on the Figure 4.7 gives one possible reason for the obtained classification results. The upper panels are similar to the Figure 3.1. However, instead of showing the spike rates used in the analysis, the spike counts are presented. Since the window of fixed size is employed, the two data representations are equivalent, but this illustration

shows the variability of the obtained feature vectors more clearly. As before, each row corresponds to one cell, and in this case, each column to one window position. The spike counts are shown for the three choices of window length, 900, 500 and 100 milliseconds from left to right. The red dashed line separates the two classes, and the non labeled rate vectors are not shown here. The non labeled examples are not used in the training phase, and does not contribute to the observation we want to illustrate. In general, they look relatively similar to the examples shown on the figure. The lower panels are the histograms calculated from the corresponding matrices, i.e. those in the same column. The x axis shows the number of spikes in a window, the value that goes from 0 to 36 in the considered set, and the y axis is the percentage of windows that contain the number of spikes given on the x axis. The color bar of the right side of the lower row corresponds to the color coding of the spike counts from the upper panels. Naturally, bigger windows contain more spikes, i.e. the spike counts increase from right to left. More important characteristic that can be seen on these figures is the lost of diversity of the evaluated spike rates. The right most matrix, calculated for $W = 100$ milliseconds, contains mainly zero values, i.e. 60% of all the values in the matrix are zero. Around 20% of windows contain only one spike, around 10% two spikes, and the remaining windows have 3 or 4 spikes. Therefore, most of the rate vectors become similar and the distance between the two classes decreases. In this particular example, the information contained in the activity of cells 3, 5 and 7 is almost lost and certain number of non-zero values is maintained only for the cells 1 and 2. Since most of the values, around 80% of them, contain only zeros and ones the rate vectors become, practically, the binary code for the classes. This is not an obstacle for itself, but becomes a problem when working with small populations. Suppose we have an information coded by a 1000 bits word. If we randomly select 10 out of those 1000 bits, we will likely lose the encoded information. Having a longer integration time is a compensation for the small population size. On the left most panel, the spike rates become more diverse, and the chances for successful classification increase.

The outcome of the last series of tests performed using this method, is shown on the Figures 4.8 and 4.9. As already described in the Section 3.3.1, in order to assess the presence of motor intention continuously along the recordings, the probability of motor intention is estimated instead of the binary class. The rate vectors are evaluated using a sliding window as before, but the training and test phases are somewhat modified. The test set is systematically chosen in advance. Every time, we will first choose a target and trial to analyze, and then select all the recordings collected for that target and trial combination. That will include recordings for all seven tasks. Each combination of trial and target is analyzed this way. The remaining data is preprocessed as before, and divided into the training and validation set for obtaining a classifier. No cross-validation is applied here, randomly chosen 20% percents of the data are assigned to the validation set, and the rest to the training set. The optimal SVM classifier is, then, applied for the rate vectors calculated from the test examples. For each of the seven tasks, one probability estimation, a value from the interval $[0, 1]$, is obtained for each window position. It indicates the presence of motor intention if exceeds 0.5. The employed SVM software package [18], includes the algorithm for probability estimation in classification tasks. The probabilities are estimated using a two-step optimization procedure. In the first step, a pair of parameters is evaluated for each rate vector example from the training set. The two parameters correspond to the probability that the considered rate vector belongs to one of the classes. The parameters are described using a fixed function, and the free parameters are chosen by minimizing the log-likelihood function. In the second step, the constrained optimization problem is solved in order to compute the probabilities out of the parameters obtained in the first step.

All the examples shown in the same figure, correspond to the same file within a database,

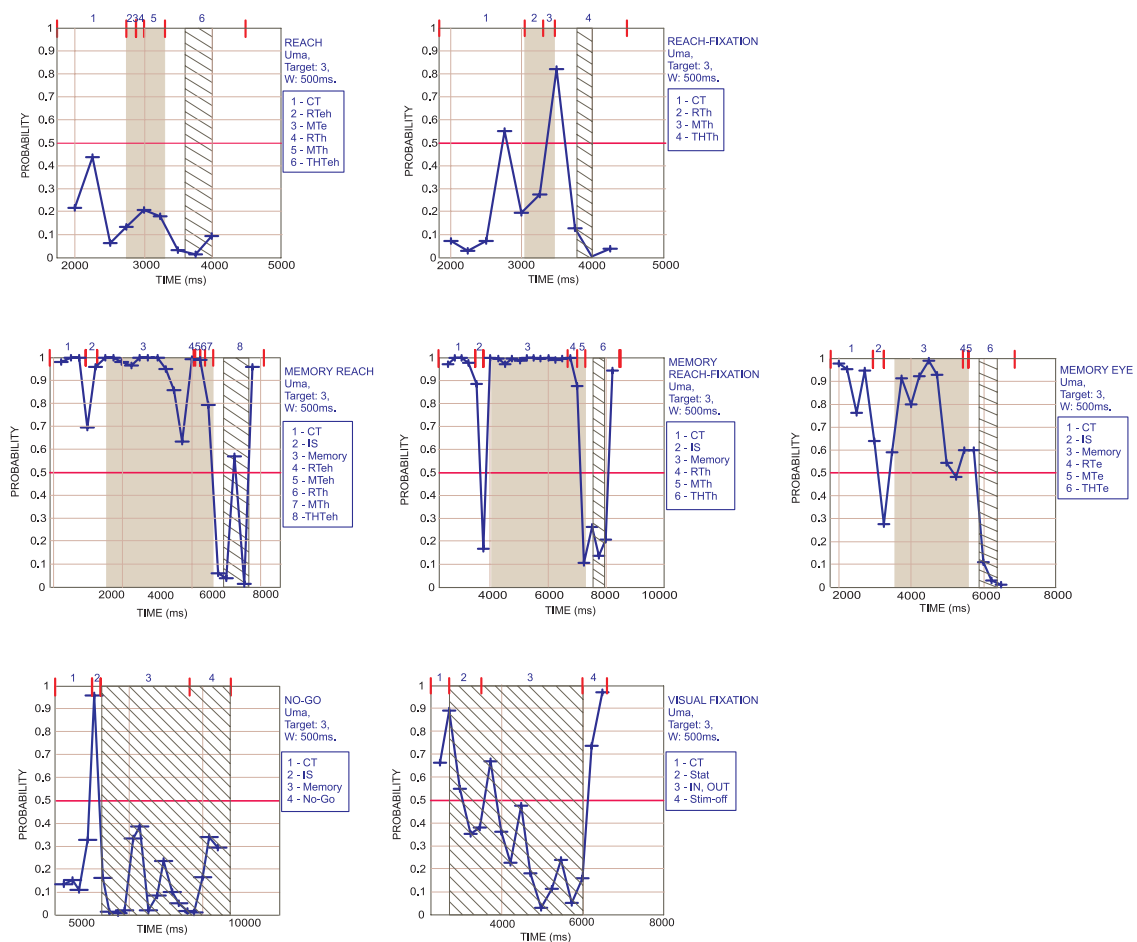


Figure 4.8: Probability of motor intention evaluated for the seven tasks, one trial, and one target position (3, on the Figure 2.7). Window size 500 milliseconds, file 'Uma_54'. First row: R (left) and RF (right); second row: MR (left), MRF (middle), and ME (right); third row: NGO (left) and VFIX (right). In grey: the time intervals with high probability of motor intention according to the corresponding experimental epochs. Hatched: the intervals where the low probability is expected following the same argument. The probability estimation is given with the central curve, each cross corresponds to one window position and is placed in the middle of the window. The threshold value for discriminating between the motor intention and no intention is indicated with the horizontal red line (at 0.5). Red bars on the top of each panel are the epoch marks. The corresponding epochs are given in the legend on the right of each panel. All the results are aligned according to the beginning of the CT epoch. At the beginning of each experiment some preparatory epochs are included, but they are not relevant for the presented analysis. Still, the time on the x axis is not aligned to CT to indicate the duration of the experiments, and the starting time of CT.

the same target position, and are recorded during the same experimental trial, from the same set of cells. Therefore, the seven panels are comparable. The top two panels in both Figure 10 and Figure 11 represent R and RF tasks. For both tasks the performance is relatively poor for the window of 500ms. This is the consequence of using relatively large window in the analysis; for these two tasks, motor intention and no intention related intervals are relatively short, and, for a bigger window, these tasks are represented with a very small number of cases

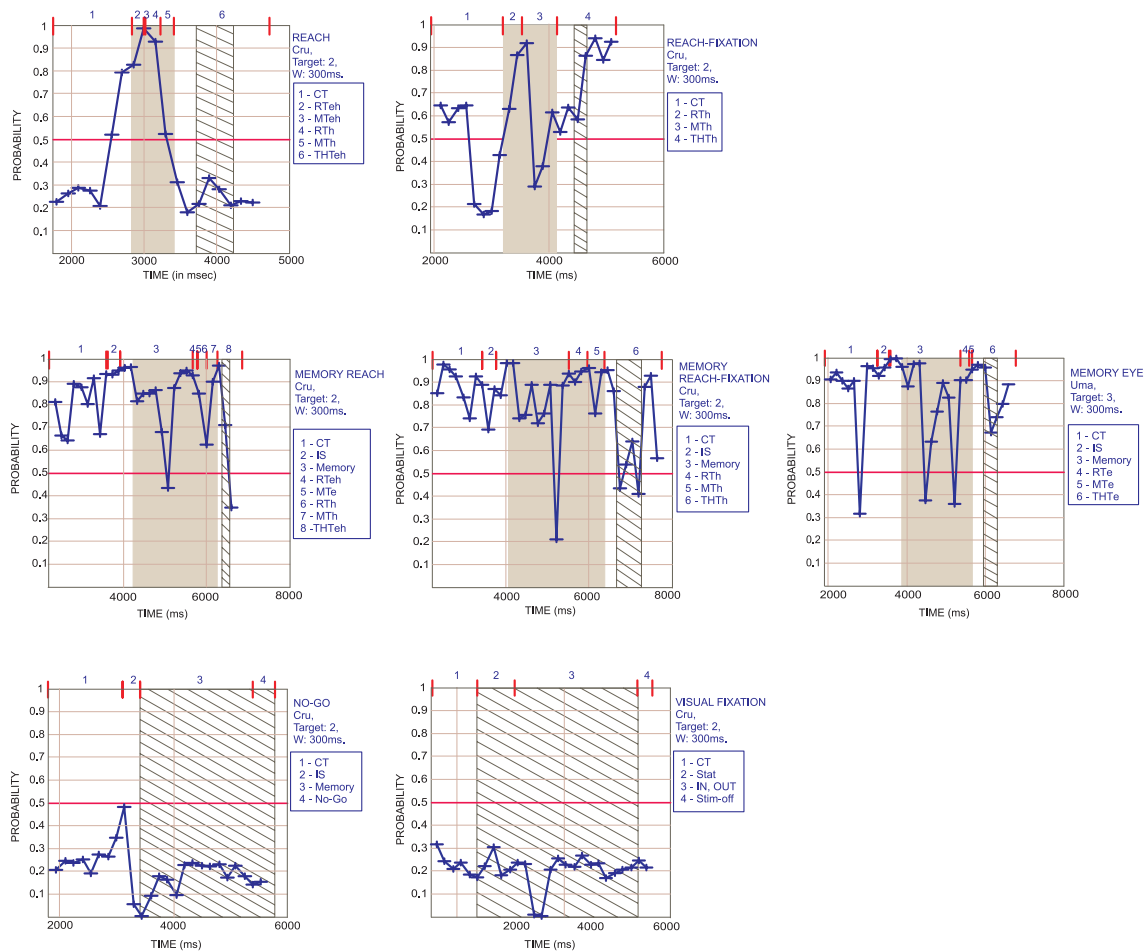


Figure 4.9: The second example of probability estimation, file 'Cru_26'. Window size is 300 milliseconds.

in the learning database. Therefore, the learning set is not informative enough for the correct classification of R and RF tasks. On the contrary, the example with the smaller window used shows much better correspondence with the expected results; the increase of probability is evident for the R task, while motor intention is not present in the rest of the recordings. For the RF task, it is more difficult to explain the result, but the peak in probability is evident in the marked interval. The three memory tasks are much easier for classification, and exhibit better results than the R and RF, for both applied windows. This is expected since MR, MRF and ME provide much more data to the training set, due to the larger number of epochs where the presence/absence of motor intention can be reliably identified in advance. Therefore, these tasks are much better represented during the training process. The obtained results are better for the longer window, which corresponds to the overall improvement of the results when longer time spike trains calculating is allowed. Finally, NGO and VFIX show good correspondence with the expected results; motor intention is present at the beginning of the recordings, then, it decays in the marked intervals, and eventually increases again at the end of the experiment.

The last issue that has to be discussed in this context is the representation of experimental epochs in the set of rate vectors obtained using this method. As shown on the Figure 2.9, the epoch lengths vary from less than 100 milliseconds (for MTe) to more than 2.5 seconds (for Memory). Sampling the recordings with a fixed window leads to unequal representation

of epochs in the training and test set. Consequently, the obtained classifier will be more influenced by certain epochs, than the others. Some of the spike rates are calculated using only a part of an epoch, while the others span several epochs. The epochs RT and MT are much shorter than Memory or THT, for all the tasks. Even for the shortest window (300ms) some of RT and MT epochs are taken together, while even a long window cannot entirely cover the Memory or NGO epochs. The number of examples is directly proportional to the corresponding epoch length, which leads to much bigger data set for bigger epochs. Typically, the Memory epoch gives rise to the majority of examples for the first class. The RT and MT epochs are less influential here, than in the first method, since they result in a relatively small number of examples; furthermore for a relatively large window, the algorithm cannot separate them from the Memory epoch. Due to the window size and the duration of the THT the latter is often excluded from the analysis; since in order to take into account the neural activity of this type a window should entirely fall into THT epoch. For the windows bigger than 500msec this never happens, while for the smaller ones, THT contributes for no more than 6% of the total data set size.

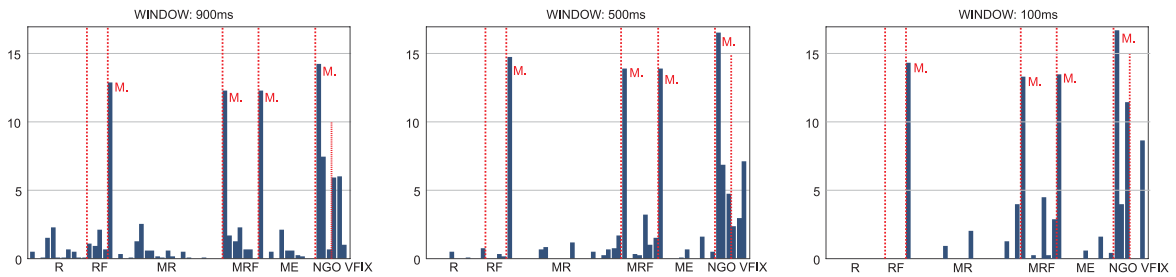


Figure 4.10: Epochs representation in the training and test set obtained from spike rates calculation using the sliding window. The examined window sizes are 900, 500 and 100 milliseconds, from left to right. x axis shows the epochs and combination of epochs obtained during spike rates calculation, y axis shows the contribution of the particular epochs or combination of epochs to the total set, for each of them the ratio between the number of corresponding examples and the total number of examples in the set is calculated. The seven tasks are separated with red lines. Memory examples are additionally marked with **M**.

The Figure 4.10 additionally illustrates these observations. The representation of certain epochs or combinations of epochs in the set of rate vectors is shown for the file 'Uma_54' and the three windows, 900, 500 and 100 milliseconds, from left to right. The examples belonging to the seven tasks are separated by red lines on the figure. The rate vectors calculated from the Memory epochs alone are marked with **M**. It is evident that they dominate in both classes, and particularly the motor intention class. Also, it is evident that THT examples are rarely present. In the no intention class, i.e. NGO and VFIX tasks, the epochs other than Memory are better represented than in the opposite class. We can conclude that in these tests, the comparison is carried, mainly, between Memory epoch in R, RF, MR, MRF, ME on one side, and NGO, VFIX examples on the other.

4.5 Discussion: Methods comparison and relevant algorithm parameters

In this chapter we presented the results obtained from the two machine learning based methods to detect motor intention from multi-electrode and multi-task recordings of neuronal activity.

The entire database consists of a list of files, and each one of them is analyzed separately, i.e. one classifier was constructed per file. The results obtained for different files were used to assess the average and the best performance for the proposed methods. The two approaches are implemented in order to examine various parameters that influence the performance. First we evaluated one vector of spike rates per epoch, in order to compare different epochs and investigate how reliable is their discrimination based on presence and absence of motor intention. In this context we assessed the single cell role in motor intention coding within a small population, and the influence of each experimental epoch. In the second series of tests, we fixed the window size and calculated the rates by shifting that window for a fixed step size. The performance as a function of the window size is examined in this context.

Precision: In the first test, the classification of epochs, the obtained error rarely exceeded 30% for the better of two databases, while for the second one this error stayed below 35%. Most of the examples resulted in the error between 25% and 35% for the first, and 20% to 30% in the second one. The best example we observed in the first database provided around 25% error, and for the second one 18% classification error

In the second method it was more difficult to estimate a general trend, since the results changed more rapidly for different files. The window length determined the range of observed errors; for the long window of 900 milliseconds the average error was 22% and 16% for the two databases, for 500 milliseconds the averages were 25% and 22%, and for the smaller window of 300 milliseconds, the average errors were 35% for both databases. The best examples from the database 'Uma' gave 5% error for $W = 900$ milliseconds, 10% for 500 milliseconds, and 25% for 300 milliseconds. The database 'Cru' contains even better examples which give less than 5% for the window of 900 milliseconds, around 6% for 500 milliseconds, and 15% for 300 milliseconds. The best file is 'Cru_26' which also gives good results for the first method, and additional examination of the included epochs and the balance in error rates for the two classes revealed no suspicious information.

Epoch contribution: The first method focuses on epoch properties and examine their role in motor intention extraction. A closer examination of errors obtained from certain epochs shows that some tasks and epochs systematically outperform the others. Typically, the precision obtained from NGO and VFIX tasks was much better than for the other examples. The examples related to these two tasks are all assigned to the same class, no motor intention. Therefore, it never happens that the same set of recordings provides data in both classes, which reduces the chance that some additional processes present during an entire recording session make the examples of the two classes similar. This can be the explanation for the relatively big percentage of errors on the THT epochs from the remaining five tasks. There, several epochs preceding THT provide data for the motor intention class, while THT corresponds to the opposite one. Still, certain similarities between these examples exist, particularly taking into account the short total duration of recordings (less than 10 seconds). This also provides an argument to use both the examples of NGO and VFIX tasks, and the examples from THT epochs. The first set of examples provides good representation of the absence of motor intention, and cannot be related to the opposite examples at the same time. On the other hand, the THT examples provide a chance to monitor the motor intention dynamics within the same recordings. Taking into account both groups, we support the statement that the extracted information indeed represent motor intention. Additionally, RF and MRF typically perform worse than the other tasks. Here, the eyes movement is forbidden, as the monkey looks at the center of the screen throughout the entire experiment, and reaches for the target with the hand only. Also, in the other tasks, we can observe very small error obtained on the MTe epoch, the eyes movement time. Also, RTh, the period of preparation for the hand movement,

provides relatively small error, but only in the R, MR and ME tasks. High classification error in the tasks that require eyes immobility, and low error in the eye movement epochs, could indicate that the particular recorded set of cells exhibit stronger motor intention in relation to eyes, than hand movements.

A potential problem in this classification method is the great variability in the epoch length. Some rates are calculated using only a couple of hundreds of milliseconds, while for the others a couple of seconds of activity was available. The Figure 4.7 illustrates the differences between the rates calculated for small windows and those obtained for the large windows. The second ones permit much bigger variability in the calculated rate vectors. It would be advisable to test the obtained results using a method that limits the variability in epoch length. For example, longer epochs can be divided into several intervals of comparable length to the smaller epochs. All these intervals can be combined to obtain the feature vectors of (number of cells)x(number of intervals) size. Then, a method for reduction of dimensionality, for example PCA, can be employed to obtain equal number of features for each epoch.

Comparison of the two approaches: As already stated in the previous section, the comparison between the two methods is not straightforward. In order to highlight the differences we need to consider the influence of different experimental epochs, included in the analysis. While the first method equally represents all the epochs, the second is biased toward the longer ones, due to the adopted sliding window procedure. As shown on the Figure 4.10 the second method dominantly depends on the rate vectors obtained from the Memory epoch. The THT epochs, the influential source of errors in the first approach, is lost here, which is likely to decrease the total error. The classification in the second method practically becomes the comparison between the Memory epochs in the R,RF, MR, MRF, ME on one side, and NGO and VFIX examples on the other side. On the other hand, the signals belonging to certain epochs are not separated anymore, as in the first approach, and many of them are represented in the same rate vectors. It is not clear what impact this has on the complexity of the classification task.

Taking into account these observations, we can compare results of the two methods. For the shorter windows, the first method is more precise than the second one. As the window increases, the second method performs better. The reason for these is twofold. First, for a bigger window, most of the data in the second method are calculated using longer segments of spike trains than in the first one. This enables longer time for integration of neural signals, which is particularly important taking into account a small population size used throughout the analysis. Also, for all the files, the second method suppresses or, for large windows, completely removes the most 'critical' examples, those belonging to the second class, but that are corrupted by the influence of preceding events corresponding to the first class. The opposite holds when using the short windows, the integration time decreases, and the 'critical' epochs are more influential.

Number of cells and single cell influence: Another issue we analyzed is the dependency of the classification performance on the number of available cells in the considered file. This is a critical parameter, that strongly influences the results. The employed multi-electrode system permits monitoring of up to fourteen cells at a time, which results in between two and eleven recorded cells, i.e. a relatively small population. Since we are concentrated on a theoretical study of the neuronal system behavior during movement planning and execution, rather than building a robust, long-lasting system the analysis of such small ensembles can give relevant results in this context. Additionally, a more detailed analysis of the population size influence is carried on the artificial model, and those results will be presented in the next chapter.

A series of tests is carried out in order to examine the influence of each cell in the set. The outcome indicates that motor intention is coded in a redundant and distributed way. Most of the cells can be removed without changing the result, or causing a change that statistical tests do not identify as significant. Removing a critical number of them (even those recognized as 'the least informative' ones among them) leads to the performance degradation. Cells that cause the statistically significant increase of the classification error are less numerous in a file. Still, this conclusion holds only within a small set of cells recorded simultaneously. Recording a bigger number of neurons will enable selection of many such cells. The best file example observed in both databases, 'Uma_54', corresponds to this description. It contains three cells that significantly increase the error when removed from the set, and five more cells whose removal does not produce change. Still, attempt of classification using the first three cells only results in much worst performance than for the complete set. Finally, it rarely happens that some cell significantly deteriorate the classification performance. We can find, more often, cells that do not contribute much, but do not disturb the computation either. This might be the consequence of the applied method robustness.

Window size: The second method permits testing the influence of the integration time length to the algorithm performance. The results obtained for all the files are consistent, the classification precision is improved when the longer integration time is allowed. The Figure 4.6 summarizes the results and presents the main trends observed. First, the lower panels show the persistent improvement obtained for the increasing window size, the performance obtained when averaging results for all the files in the database. The upper panels illustrate this tendency for each file separately. For the majority of examples the error decreases for the increasing window, already when comparing the results for 900 milliseconds and 700 milliseconds window this is evident. The difference is even more visible when comparing them with the 300 milliseconds window results.

Probability of motor intention: The last series of tests estimates the probability of motor intention along the recordings. The recordings collected for seven tasks, for one fixed trial and target position are removed from the set before the training to be used in the test phase. The outcomes are presented on the Figures 4.8 and 4.9. It is difficult to evaluate the results obtained in these tests, but some mismatches with respect to the expectations can be observed. The results obtained for the R and RF tasks in the first presented file does not follow the expectations, while in the second file they are somewhat better. The reason for that is given on the Figure 4.10, which shows tha R and RF are poorly represented in the set of rate vectors, due to the short duration of epochs of interest. For the MR, MRF and ME tasks the results are better, although some mistakes, particularly for the THT are present. As before, the NGO and VFIX demonstrate the best performance.

Comparison with the results from literature: At the end, it is important to discuss the differences and similarities between this approach and the related studies presented in the literature. The problem of motor intention decoding can be related to the state estimation described in [84, 79, 65, 2]. The first three of them, also analyze the parietal cortex activity, and the last one uses the recordings collected in the premotor cortex. These studies compare the typical neural activity observed in certain experimental epochs, and develop methods to distinguish between the related states of the movement. The transitions between the states can be used to generate 'go' and 'stop' signals for the BCI. In our case, the transition between the motor intention and its absence can be employed in the same context. It is important to emphasize the major difference between the approaches. These studies aim to distinguish between different epochs, and all the example of the same class are derived from the same epoch. On contrary we examine the similarities as well as differences between certain epochs

in our classification task. All the epochs (or states) are divided into the two groups, and difference between the two groups as well as the similarity between the examples in the same group determine the algorithm precision. The examples from the same class could form several clusters, corresponding to certain epochs, but these clusters have to be sufficiently different from the ones belonging to the opposite class.

Most of these studies use the state classification to improve the precision of movement direction estimation, and the obtained precision is reported in that context. In [84] the activity of the states equivalent to our CT, Memory and RTh epochs is demonstrated, and the three protocols for state and target estimation are proposed. The method comparable to our study required recognizing 500 milliseconds of Memory activity to release a 'go' signal, and the state classification precision was almost 100% when using 40 cells, and around 90% for 10 cells. Also, [79] reports 10% error in the classification of five states when using 50 units. The same precision is obtained for 20 LFP channels, but this result cannot be compared to our study. In [2], the precision of state classification is not reported alone, but in combination with target estimation. The four strategies are proposed and the classification precision together with the latency is reported for them.

The study presented here demonstrates the motor intention identification with approximately 25% to 35% error for one and 18% to 30% for the other database, using the epoch classification presented before. The obtained results are inferior to the numbers presented in other studies, but several differences have to be taken into account in this comparison. The difference related to the problem definition is already stated, instead of classifying neural activity collected during certain epochs, we aim to observe the similarity between several epochs that we refer to as motor intention. Additionally, the results from the literature demonstrate a sufficiently big difference between certain epochs like CT, Memory, MTh, MTe or CT, Memory and RTh that can be detected by a classifier. In our work, we consider Memory, RTh, RTe, MTh and MTe in the same class, in spite of the evident differences, in order to examine whether all of them share the same information about the motor intention. Consequently, the problem proposed here is more difficult since it attempts to identify just one information encoded in the data out of many others present during a reaching task. The study of error distribution across experimental epochs shows how well each of the included epochs contributes to motor intention encoding.

The number of cells used in classification also plays an important role, as already described. The [84] reports 10% error when estimating three states (CT, Memory and RTh in reaching tasks) from only 10 cells, and [79] states that 50 units are needed for the similar prediction of five states, CT, Memory for reach, Memory for saccade, MTh and MTe. In our study we combine all five listed epochs, together with the THT epoch from reaches and saccades, and the four additional tasks. Therefore, it is possible that we need more than 50 cells for obtaining a comparable performance, the assumption that cannot be tested due to the limited size of considered populations. The second referred study, [79], combines the recordings obtained sequentially, from different positions and recording sessions, in the same task. We adopted a different approach, to use only the data recorded simultaneously in the same decoding session. The alternative approach can be tested here as well, but it does not guarantee a better performance. The motor intention considered here is a high level neural signal, and it is not sure whether we can compare its influence on the neural activity obtained during different sessions, performed in different days.

The integration time used for calculating rates is also an important parameter in these studies. Its influence is examined in details in the context of target position estimation, but the conclusions can be extended to extraction of any motor parameter from the neural recordings. The [77] demonstrates the increase of classification precision for around 25% when increasing the window size from 250 to 450 milliseconds. There, the neural activity from the Memory

epoch is used to predict movement toward one of eight targets. The increase in classification precision for the increasing window size is also demonstrated here, and presented on the Figure 4.6. It is shown that relatively long windows should be used in order to obtain good precision. Some of the literature propose using 100 milliseconds [100, 16] or 200 milliseconds [2] windows for calculating rates from the activity of various cortex regions, although the results using only 20 milliseconds for the motor cortex cells are reported [39]. The studies related to the parietal cortex use 250 and 500 milliseconds activity [84], and [65] used 900 milliseconds, the constraint imposed by a small number of cells. Here, only 8 highly tuned cells are employed for classification. This number is comparable to our study where we used up to 14 cells, but they are randomly chosen which is an additional disadvantage. The employed window of 900 milliseconds gave good results in our study as well.

The presented analysis is done in an open-loop manner since the entire database was recorded prior to the analysis. The report [16] shows the reorganization of cortical activity in the closed-loop tests with respect to the open-loop ones. The closed-loop stands for 'brain control' tests where the cursor position is controlled directly from cortical activity, and the open-loop stands for the off-line estimation of target/trajectory from the recordings collected while the monkey performed natural movements. The paper [83] demonstrates the successful trajectory estimation from small populations using linear filters, the result explained by neural activity adaptation to the closed-loop tests. Therefore, we may hypothesize that repeating the presented study in a closed-loop tests can improve the performance even for a smaller recorded population.

Finally, the number of trials in the training data set should be compared. Most of the methods for target estimation use only a segment of neural activity, often from Memory epoch. This way, every experimental trial results in one example in the training set for the classifier. In the state estimation studies, several segments of the same trial corresponding to different epoch are employed, which is more similar to our approach. In [65] the first 30 trials in each daily session were used for training of a target classifier, while [78] uses 80-100 trials for each of the studied conditions. In [84] 8-16 trials per task and direction are executed. In this work, as already illustrated on the Figure 2.10, the first six tasks are executed four times for eight target positions, and the seventh one three times for sixteen targets. In total, it results in 240 trials in the same file, 80% of which is used for training. The number of trials per experimental condition is somewhat smaller in this study than in the examples from the literature. In general, we combine many experimental conditions, tasks and epochs, in the same classification task, and it is important to represent each of them well in the training set. Increasing the number of trials per condition can improve the overall classification performance.

The general conclusion of this study is that the proposed classification problem, namely, distinguishing tasks, as well as distinguishing epochs within the same recording session, based on the presence/absence of intention for making a movement, can be solved by a traditional SVM algorithm with a satisfactory performance. In general, a bigger set of simultaneously recorded spike trains gives a better performance, but the relation between the number of recordings and the algorithm performance is not straightforward. We interpret this observation by the different roles in information processing played by different neurons. Some neurons play a more important role in encoding motor intention than others. Spike trains from those neurons are more efficient in building classifiers by machine learning methods than spike trains from other neurons. On the physiological side, the assumption that posterior parietal cortex encodes motor intention finds additional support from the results of this study.

Chapter 5

Artificial neural network

In this chapter we present an artificial neural network constructed to mimic some aspects of the experiments described in the first chapter. The model represents a moderate size network of leaky integrate and fire neurons, constructed following the recurrent neural network, or echo state network paradigm. The network inputs mimic the experiments presented in the first chapter, and the output spike trains are used in the similar classification procedure as the experimental data. First, the two selected experiments, RF and NGO, are modeled by a set of nine signals representing the eight targets and the central fixation light. They are converted into regular spikes and used as the inputs into a network that can 'memorize' the instructions for a limited period of time. The evoked spiking activity from the layer of neurons is analyzed similarly as the data collected during the experiments on monkeys. The goal was to identify the presence and absence of activity following the combination of input signals that indicate a movement. In the context of this model we call it the 'motor intention', although a relation with the signal obtained from the experimental data cannot be established for such a simplified model. The aim of this study is not to evoke motor intention in an artificial network using the appropriate combination of inputs. The mechanisms that create motor intention in the cortex, or in its parietal region are not completely examined and claiming that a limited knowledge combined with a limited size model produces the same signal as the complex biological system would be very hypothetical. Instead, our goal was to formulate a simple model which behavior resembles that of the biological system and can be used to test some relevant parameters listed in the previous chapter. Precisely, we want to demonstrate the improvement of the classification precision when the number of considered artificial neurons increases. This question has arisen as a consequence of decoding motor intention from small sets of simultaneously recorded spike trains. The results presented here support the intuition that the same method applied for sufficiently big populations result in good classification of the signal of interest. Also, the influence of integration time used to calculate spike rates is assessed.

This chapter is organized in the following way. In the Section 5.1 some basic facts regarding the RNN theory are listed, together with some examples of modeling parietal cortex on different level of details, reported in the literature. The Section 5.2 describes the building blocks of the model, the neuron and synapse models, together with the network description, emphasizing the parameters of interest. An important step in building the model is the formalization of the selected experimental tasks described in the Section 5.3. In the Sections 5.4 and 5.5 the procedure for designing appropriate networks is described, and the classification task and results are described in the Section 5.6. Finally, we will conclude with the Section 5.7.

5.1 Background: RNN and neuronal network models

At the beginning of this chapter a short description of the concept of recurrent networks (RNN) and liquid state machines (LSM) is presented. This computational paradigm was introduced in the context of spiking networks in [54, 53, 42, 37], while [40] develops the same concept for the networks with analog nodes, so called echo state networks (ESN). We focus on the first type of networks, that will be implemented in the model studied here. Several characteristics of the LSM make it a good model for the neural computations in the brain. The real-time computing is feasible here, due to the 'liquid states' paradigm. In other words, the network is not required to reach a stable state, or to switch between stable states in order to complete the certain task. Instead, it is constantly in transition, following the input signals. This enables fast computing that can be related to the functions of the cortex. Another characteristic that relates the LSM with the cortex is the parallelism of computations. It is evident that the brain can perform a huge amount of operations in parallel, and under certain conditions the LSM can execute multiple operations. The mentioned condition is first recipe for generating such networks, the imposed connectivity structure among the nodes has to be sufficiently sparse to enable diversity inside the network. Having strong coupling in the system will lead to synchronization of some of the network nodes, effectively decreasing the number of computational units able to contribute to different tasks. Finally, the short-term memory is typically observed in these networks. The input signals affect the adjacent nodes, and they convey the information further through the network. Each node represents a dynamical system of certain complexity, and its internal states become affected by the input signals. This way, the inputs are 'memorized' in the network states for a certain period of time, dependent on the network parameters choice.

Another specific issue related to the LSM and ESN is the adopted training procedure. While the classical machine learning theory applied for the recurrent neural networks proposes the complex and computationally demanding backpropagation algorithm, here the training is reduced to solving a linear regression problem. Instead of an additional modification of the network parameters to fit the data, we assume that the network nodes modify their states according to the network inputs, and since the information about the inputs is already stored, the remaining task is to tune the outputs to represent the input-output function properly. Therefore, the training procedure is reduced to solving a linear regression problem, and the precise explanation of one such task will be presented in the Section 5.5.2.

Several implementations of the LIF model are reported in the literature. In [37] a relatively detailed model of a cortical column is implemented using the LIF concept, while in [42] somewhat different task is posed for a similar system. The LIF is generated and trained to mimic the motor cortex and operate a 2-joint robot arm.

Although we cannot provide any biological explanation for the model, nor assign it to any particular cortex region, it is useful to refer to several prominent studies of the parietal cortex mechanisms through the models. The examples that will be listed here should illustrate the variety of models used to explain particular processes and characteristics observed through the experimental work on the parietal cortex. We will not attempt to provide an extensive overview of such studies, the only motive behind this is to illustrate several attempts to explain experimental observations in the parietal cortex through appropriately constructed models.

In [75] a relatively simple model of network is used to illustrate a possible mechanism that leads to gain field generation phenomenon present in the parietal cortex. A spike rates model for neurons, or rather small neuronal ensembles, is implemented. Each connection between two neurons is represented as a combination of Gaussian functions that depend of the preferred directions of the connected cells. This study demonstrates the rise of gain field in a network

of this type, hypothesizing that the recurrent connections in combination with directional tuning generate the phenomenon of interest.

One another study [102] attempts to explain the integration of sensory inputs of various modalities in the parietal cortex through the appearance of gain fields. The experiments illustrate that this region is influenced by various signals, like eye-related, hand-related, audio and other signals, that are initially coded in different coordinate systems. An interesting question arising from the experimental studies is how are the inputs transformed from one to another coordinate system, and in which system they are represented in the PC. This study analyzed several models consisting of three layers and receiving signals in eye-centered, head-centered or body-centered coordinates. The report analyzes and compares the differences in models evoked by different inputs.

Unlike these two studies using the minimal description for the network elements, in [23] a detailed model is presented. This report examines the influence of the synaptic receptor, NMDAR and AMPAR and GABA_ARs, to several processes in the parietal cortex, like population directional tuning, memory and influence of distractors, and others.

To summarize, the researchers proposed various models to study the processes observed in the parietal cortex. The level of details in the models goes from a simple rate model, to the complex Hodgkin-Huxley model, and from static synapses described by a single constant, to biologically plausible synapse models. The general principle was to implement as simple as possible model that can still manifest the property of interest.

In this report we adopted the LSM for two reasons, related to the parameters which influence we wanted to examine. First, for studying the population size effect, we needed a network of neurons that communicate among themselves but exhibit different behavior in relation to the examined problem. For assessing the window size influence, we needed a spiking network which possesses a certain memory. For a network that can solve the given problem in one step, there is no need to integrate outputs over a certain time interval.

5.2 Building elements

In this section the elements of the adopted model are described, and the corresponding notation, used throughout the entire chapter, will be introduced. A simplified network model is implemented in this study for several reasons. First, we aim to model some functional properties of the cortex region, and the obtained neural activity during reaching and saccadic movements. The biological system, the cortex network, involved in such tasks is extremely complex, and although its certain characteristics are known many other mechanisms are still an object of experimental studies, as described in the first chapter. We do not aim to assess all its properties, but just some of its functionality relevant in this context, in order to test the parameters that cannot be analyzed otherwise. The experiments representation is also very simplified and contains only the basic feature of the real experiments. Since we already approximate many characteristics of the system of interest, and the input signals, there is no reason to implement detailed model for cells and synapses. For cells, we use the leaky integrate and fire (LIF) model, and for the synapses pulse coupling that includes the synaptic weight, the propagation delay, and the rise and decay times for the pulse.

5.2.1 LIF neuron model

The leaky integrate and fire neuron is implemented according to the expressions given in [35, 24]. It captures only the cell membrane potential dynamic of the real neurons, and does not include any description of the ion channels or ion fluctuation through the cell membrane.

Therefore this model can produce action potentials but does not contain details of the mechanisms behind it. Mathematically, it is described by one dynamic equation for the membrane potential, and the membrane resetting condition. Such neuron models, described by a single variable, are also called the single-compartment models. The entire cell membrane is represented as a parallel connection of a resistor and a capacitor, and the mathematical expression for that is:

$$c^m \frac{du(t)}{dt} = -i^m(t) + \frac{I^e}{A}.$$

where c^m represents the membrane capacity, $u(t)$ is the membrane potential, $i^m(t)$ the membrane current expressed as the current per unit area, I^e the external current divided with the total surface of the neuron A . In the LIF model i^m is a single passive leakage term, a function of membrane potential. Therefore, the equation becomes

$$c^m \frac{du(t)}{dt} = -\frac{u(t) - u^r}{r^m} + \frac{I^e}{A}.$$

where r^m is the membrane resistance and u^r the resting potential. Finally, the same equation can be expressed in the form that will be further used in this study

$$\tau^m \frac{du(t)}{dt} = -u(t) + u^r + R^m I^e \quad (5.1)$$

$\tau^m = c^m r^m$ is the membrane time constant and $R^m = A \cdot r^m$ the total membrane resistance. The membrane potential can take any value from the interval $[u^r, u^{th}]$, i.e. between the resting and threshold potential. When it reaches the threshold value, an action potential is generated and the membrane potential is returned to its resting state, the condition formally expressed as:

$$\text{if } u(t^*) = u^{th} \rightarrow u(t \in [t^*, t^* + \tau^r]) = u^r, \quad t^f = t^*. \quad (5.2)$$

The spike times will be further denoted as t^f throughout this text. The time constant τ^r represents the refractory time, the interval of time after the release of an action potential while the cell remains in the resting state, and the membrane potential cannot increase.

For a constant input current $I^e = \text{const.}$ and assuming a cell in the resting state at the time $t = 0$, the membrane potential as a function of time can be evaluated analytically

$$u(t) = u^r + R^m I^e \left(1 - e^{-t/\tau^m}\right)$$

and the time of the next action potential is given as $t^f = -\tau^m \ln \left(1 - \frac{u^{th} - u^r}{R^m I^e}\right)$.

5.2.2 Synapse model

Following the same argument as before, the model for synaptic coupling is also simplified and so called pulse-coupling is adopted. There, the only parameters included are the synaptic weight, the propagation delay, and the pulse rise and decay time constants for each action potential, as described in [35].

The total input current for a cell i is given with the expression:

$$I_i(t) = \sum_{j,f} w_{ij}^f \cdot \alpha_{ij}(t - t_j^f) \quad (5.3)$$

This function is obtained as a summation of all the synaptic currents coming to the cell i from the cells connected to it. The time of the f -th spike for the cell j is denoted as t_j^f ,

and w_{ij}^f is the synaptic weight at the time t_j^f . The function $\alpha_{ij}(t - t_j^f)$ approximates the post-synaptic current increase as a result of the incoming spike at t_j^f from the cell j , and it is given as

$$\alpha_{ij}(t) = \frac{q}{\tau_{ij}^{s,1} - \tau_{ij}^{s,2}} \cdot \left[e^{-\frac{t - \Delta_{ij}}{\tau_{ij}^{s,1}}} - e^{-\frac{t - \Delta_{ij}}{\tau_{ij}^{s,2}}} \right] \cdot \Theta(t - \Delta_{ij})$$

The model parameters are the pulse rise time constant $\tau_{ij}^{s,1}$, the decay time constant $\tau_{ij}^{s,2}$, the synaptic propagation delay Δ_{ij} , and q , the normalization constant calculated to obtain 1 as the maximal value of $\alpha(\cdot)$. The notation $\Theta(\cdot)$ is a Heaviside function which indicates that for a presynaptic spike at t_j^f , the postsynaptic potential appears at $t_j^f + \Delta_{ij}$.

The last expression can be simplified if we assume that the two pulse constants, the rising and the decay time, are similar and replace them with one constant τ_{ij}^s . The expression used in the model presented in this section is:

$$\alpha_{ij}(t) = q \cdot \frac{t - \Delta_{ij}}{(\tau_{ij}^s)^2} \cdot e^{-\frac{t - \Delta_{ij}}{\tau_{ij}^s}} \cdot \Theta(t - \Delta_{ij}). \quad (5.4)$$

The notation w_{ij}^f in the Equation 5.3 indicates that the synaptic weight can be a function of the previous spikes. According to the conclusions presented in [66, 55], and synaptic model implemented in the network simulation tool in [67], this parameter can change as a function of all the spikes appearing at the synapse until the considered moment in time. This should not be mistaken for the long-term synaptic plasticity which should be additionally modeled if needed. The model, named dynamical synapse, increases the variability of synaptic inputs and corresponds better to the experimental evidences. As described in [55, 96] instead of using the constant synaptic weight w_{ij} , i.e. the constant contribution of each presynaptic spike to the cell membrane potential, this parameter is the function of all the previous transmitted spikes up to the time t^f , and defined as

$$w_{ij}^f = w_{ij} \cdot U^f \cdot R^f$$

Here, w_{ij} represents the absolute synaptic efficacy, i.e. the synaptic weight in a situation when all the 'resources' of a connection are active and can contribute to the increase of the cell membrane potential following a presynaptic action potential. The function U^f describes a fraction of available 'resources' responding to the presynaptic spike at t^f , and depends on the previous spikes as

$$U^f = U^0 + U^{f-1}(1 - U^0)e^{-\Delta t^f / \tau^U}$$

where U^0 gives the fraction of 'resources' reacting to the first spike in the set, $\Delta t^f = t^f - t^{f-1}$ is the inter spike interval, and τ^U the decay time constant for U^f . Finally, the R^f is the fraction of the synaptic 'resources' remaining available after the presynaptic spike at time t^f , and depends on the previous presynaptic spikes as

$$R^f = 1 + (R^{f-1} - U^{f-1}R^{f-1} - 1) \cdot e^{-\Delta t^f / \tau^R}.$$

where τ^R represents the recovery time after the inactivation caused by a presynaptic action potential.

In order to keep the model as simple as possible, all the synaptic weights are constant in our model. The additional complexity introduced by dynamical synapses (i.e. two dynamical equations for each connection) will not contribute significantly to our model, already simplified

in many aspects. Further more, the effect we want to demonstrate can be observed in the system with constant weights, without using the more realistic synapses. We will keep the notation w_{ij} in the following sections to emphasize the use of constant weights.

5.2.3 Network

The entire network consists of N cells, and each cell i is described by the following expression:

$$\tau_i^m \frac{du_i(t)}{dt} = -u_i(t) + R_i^m \left[\sum_{j,f} w_{ij} \alpha_{ij}(t - t_j^f) + I_i^e(t) + I_i^n(t) + J_i \right] \quad (5.5)$$

In addition to the synaptic current, the direct external current $I^e(t)$ and noise $I^n(t)$ are included. The last term J_i represents the internal current in the model which provides a slow spiking of the cell, even in the absence of synaptic and other inputs.

Therefore, the entire network as a collection of cells is represented as $\mathcal{N} = \{\mathcal{C}_i\}_{i=1..N}$. Each cell \mathcal{C}_i with all its input currents can be described by one dynamical equation and a set of parameters

$$\mathcal{C}_i = \left\{ \tau_i^m, R_i^m, \mathcal{S}_i, I_i^e(t), I_i^n(t), J_i, \tau_i^r, u_i^r, u_i^{th} \right\} \quad (5.6)$$

where τ_i^r represents the refractory time, and u_i^r and u_i^{th} the resting state and the threshold potentials, as before. The set of incoming synapses to the neuron i depends on the parameters

$$\mathcal{S}_i = \{w_{ij}, \tau_{ij}^s, \Delta_{ij}\}_{j=1..N}. \quad (5.7)$$

5.3 Formalization of the behavioral tasks

Before describing the implemented model in detail, it is important to explain the model inputs and outputs derived from the original experiments presented in the first chapter. As already stated, this model aims to mimic some relevant aspects of the experiments and, following that requirement, its inputs are created to resemble the control signals in the experiments. The minimal subset of tasks needed for modeling the signals of interest is employed, only two tasks are defined and presented at the network input.

Input signals: Two input-output protocols are defined, one that corresponds to the experiments involving a movement, and one for the tasks that forbid every movement. In other words, the two contrasted tasks are included, one that should demonstrate our substitute signal for 'motor intention', and the other that exhibits no such signals. As already mentioned, the nine model inputs are available, eight for the target positions and one for the central light/go signal.

The illustration of the two protocols is given on the upper panels on the Figure 5.1. On the left task, the red pulse represents the 'go' signal, and the succeeding black one demonstrates the target direction. The green pulse at the bottom of the panel is the desired output, the 'motor intention' that should appear shortly after the target presentation. The network should memorize the instruction to move, and reacts immediately after detecting the pulse on any of the input channels for targets. On the left panel, the red pulse is not presented at the beginning of the trial, instead, the target pulse appears alone, and it should not evoke the increase in the output activity. The lower panel presents a series of typical input signals, the red CT channel is the 'go' signal as before, and the eight targets are presented in the remaining eight rows. The beginning and duration of the pulses is randomly chosen, according to the

procedure that will be described later, in order to impose enough variability in the training and test sets. The interval between the two successive trials have to remain within certain limits.

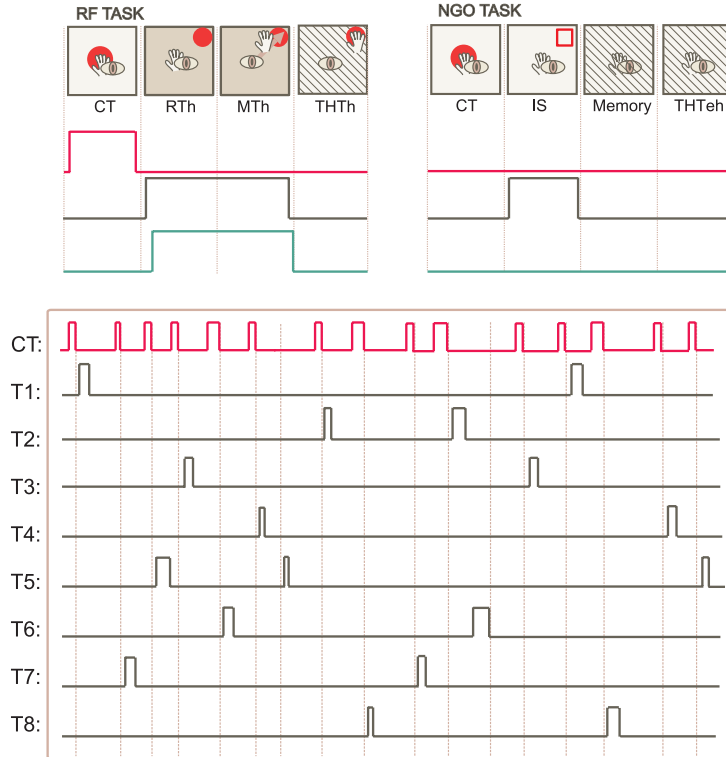


Figure 5.1: Input-output protocol and typical inputs to the model. Top - the equivalent experimental tasks presented along with the set of signals defining the two protocols. Left panel - red line is the 'go' signal, it should be memorized until the target signal (black) appears. The green output, the 'motor intention' is shown on the bottom of the panel. Right - the 'go' signal is absent and the target appears alone. This combination should not generate the increasing output activity. Central panel - a typical series of input signals, top row is the 'go' signal (marked as CT), and the inputs corresponding to the eight targets are given below.

Tasks: The two described protocols can be compared to the RF and NGO tasks. The first of them, RF, is selected among the tasks that require a movement, while the NGO represents the absence of movements. The choice of RF among the five candidate tasks is motivated by several arguments. As before, we kept the elements of the model as simple as possible but still sufficient to demonstrate the characteristics of interest. Following this principle, the protocol that initiates a movement needs only three segments - the interval of waiting for the instructions, the 'go' signal followed by the movement execution, and the final waiting phase. The type of movement executed after the 'go' signal is irrelevant here. An abstract concept of movement, the motor intention, is required in this context rather than the actual effector activity. The initial step in the protocols can be related to the CT, while RTh and MTh correspond to the interval after the end of the 'go' signal and the beginning of the target signal. The THTh corresponds to the end of the target signal and the decrease in the output activity. In the second protocol, the target presentation can be related to the IS signal in NGO, and the following time interval corresponds to the Memory and THTh.

At the end, several observations related to the choice of input-output protocols should be emphasized. Although we defined the first pulse as the 'go' signal and the second as the target position, the network likely 'perceives' them in the opposite way. The first pulse is the information about the 'movement' that should be memorized, and the target signal serves as the instruction to 'go', i.e. to increase the output activity. Although these inputs and the original experiments can be related, as already described, some differences are evident. In the experiments, the central light indicates the beginning of the task followed by the target appearance. This sequence is presented for both, RF and NGO tasks, not just the first one. What distinguishes them is the color of the presented target, that indicates the executed task, i.e. the required activity after the target presentation. Indicating the color in this context is not practical, instead, the information of interest should be indicated by the presence or absence of certain signals. Among the two available signals, the CT and the target, we have chosen to alter the first of them. Using the target as the 'go' cue will not require any memory for correct execution of the tasks, the summation of all the target inputs will be sufficient for generating the right output. Still, the presented task is not particularly complex, but is sufficient for demonstrating the effect of interest.

Finally, the justification for using a network in this context should be presented. Clearly, we needed a network in order to analyze the influence of the population size, but some requirements for the task presented to the network have to be satisfied. In other words, a single cell should not be sufficient for reproducing the input-output dependency and a network should be needed. Here, a single cell is theoretically able to solve the task if we allow any choice of model parameters. Therefore, we set the parameter τ^m in such a way that a single cell cannot do the task, i.e. the cell memory is not sufficient for generating the right output.

5.4 Model and implementation details

The complete model is presented on the Figure 5.2, the series of pulses illustrated on the Figure 5.1 is brought to the first layer of the network, denoted as 'conversion of signals' on the figure. Its outputs are further transmitted to the hidden layer, the recurrent network. Model output on the figure represents all the spike trains collected from the neurons of the hidden layer. They represents the 'recorded database of spike trains' for this model. They are first examined to chose the best generated models, and then, the outputs of those models are tested using the same methods as described for the experimental data.

5.4.1 Input layer

The first layer represents a simple translator from the pulses to the regular spike trains, that are more suitable as the inputs into the main network layer. The nine excitatory LIF neurons are placed in the input layer, each converting one of the input signals. The cells of this layer are described using the expression (5.1) for the LIF neurons, and the set of parameters

$$\mathcal{C}_i^{in} = \left\{ \tau_i^{in}, R_i^{in}, w_i^{in,e}, I_i^e(t), \tau_i^{r,in}, u_i^{r,in}, u_i^{th,in} \right\}, \forall i = 1..9. \quad (5.8)$$

These parameters are set to produce regular spike trains with frequency close to the theoretical maximum when a current is present at the input.

The inputs in form of spike trains are further transmitted into the recurrent network of the middle layer. The synapses between these two layers are given with the equation (5.4) and the parameters

$$\mathcal{S}^{in,RNN} = \left\{ w_{ij}^{in}, \tau_{ij}^s, \Delta_{ij} \right\}_{i=1..N, j=1..9}. \quad (5.9)$$

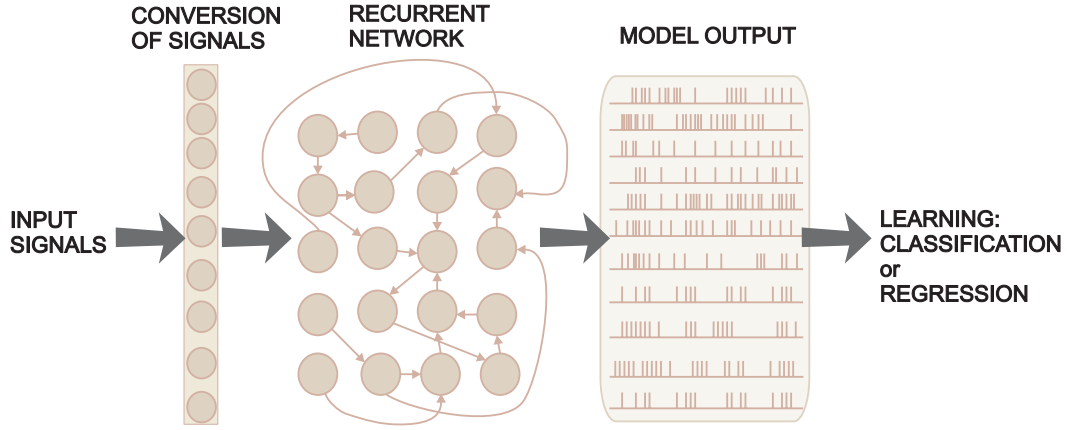


Figure 5.2: The scheme of the model: the first layer (left most) converts the input pulses into spike trains, the recurrent network of LIF neurons (middle) memorizes the inputs and produces a collection of spike trains on the output. Those spike trains are used in the classification tasks similar to those performed for the experimental data.

5.4.2 Hidden layer/RNN

The second layer represents a recurrent neural network (RNN) that exhibits a short-term memory property necessary for solving the presented problem.

The total set of N neurons is divided into N_E excitatory and N_I inhibitory cells. Each of them is modeled as a LIF neuron, but the parameters are different. The nodes are interconnected in the random fashion, using the pulse coupling model described before. The network receives inputs from the first conversion layer, and its outputs are used for estimating the network 'quality' according to some criteria that will be described later, and for examining the analysis methods used on the experimental data, as well. The set of parameters that defines the layer includes the two types of cells, the excitatory \mathcal{C}^E and the inhibitory \mathcal{C}^I ones. The recurrent connections inside the hidden layer can be excitatory-excitatory, inhibitory-inhibitory, or between an excitatory and an inhibitory cell. The choice of the parameters depends on whether they transfer signals from excitatory or from inhibitory cells. The incoming connections from the input layer are already described in the previous section, and the output connections depends on the analysis method and will be described later. The parameters defining the model are

$$\mathcal{C}_i^x = \left\{ \tau_i^x, R_i^x, \mathcal{S}_i^E, \mathcal{S}_i^I, \mathcal{S}_i^{in,RNN}, I_i^n(t), J_i^x, \tau_i^{r,x}, u_i^{r,x}, u_i^{th,x} \right\}, \quad (5.10)$$

$$\mathcal{S}_i^x = \left\{ w_{ij}^x, \tau_{ij}^{s,x}, \Delta_{ij}^x \right\}_{j=1..N_x}, \quad x \in \{E, I\}, \forall i = 1..N_x. \quad (5.11)$$

A particularly important point in developing this type of networks is the imposed connectivity structure. As proposed in the literature, the sparse matrix of connections is required to ensure a sufficient diversity inside the network. Here, each node is connected to around 20% other, randomly chosen, nodes in the network.

5.4.3 Output

The linear output layer was first used with the described model in order to test the network performance in a rigorous fashion and defines the criteria for the network 'quality'. The eight

outputs are calculated from the activity of the hidden layer, one for each target position using the expressions

$$u_i^{out} = R_i^{out} \sum_{j,f} w_{ij}^{out} \alpha_{ij}^{out}(t - t_j^f), \quad (5.12)$$

$$\alpha_{ij}^{out}(t) = q^{out} \frac{t - \Delta_{ij}^{out}}{(\tau_{ij}^{s,out})^2} e^{-\frac{t - \Delta_{ij}^{out}}{\tau_{ij}^{s,out}}} \Theta(t - \Delta_{ij}^{out}) \quad (5.13)$$

The parameters defining the layer are given with these equations.

5.5 Tuning and selection of networks

So far, the model is defined as a three layer network with nodes in the form of LIF neurons, and synapses modeled as the pulse coupling. The set of free parameters for each of the layers is just listed in the previous section, and the full description of the model is given in the Appendices (A.2, A.3, A.4). Most of these parameters are defined in advance, in the process of network initialization. They are either fixed values, or are chosen from a uniform or Gaussian distribution (depending on the parameter), with the properly chosen parameters. This corresponds to the liquid state machine paradigm for training the RNN. According to that, the parameters of the hidden layer are selected in advance, in such a way that the network posses some generally good properties. No training in the hidden layer is performed. The only coefficients calculated in the training step are in the output layer. The choice of the input and hidden layer parameters is described in details in the Appendix A.5, according to the procedure proposed in [14]. Here, we will describe the procedure for selecting the appropriate networks for further tests corresponding to those described in the previous chapters.

The software used in the presented study is either written in Matlab, or as a combination of the Matlab code and the functions provided in the publicly available software package CSIM, described in details in [36], and available at [1]. All the listed equations are discretized with the 1 millisecond time step, using the Euler method for numerical integration.

5.5.1 Randomly generated networks and selection criteria

As already described, the majority of network parameters are set in advance, according to the procedure described in the appendices. This design method aims to develop networks with good properties in general, i.e. with enough capacity to learn various tasks. The key step in this process is the choice of internal connectivities in the network of the hidden layer. Although the literature proposes some strategies for improving this choice we will use the randomly selection of the connectivity structure in this study. As described, the only requirement for the network is a sufficiently sparse topology. Every node in the network has to be connected to around 20% of other nodes, that are chosen randomly from the network. For examples, in a network of 500 cells, each of them will have 10 connections. This way, the obtained topology is random, but sufficiently sparse to ensure diversity in the dynamics of the nodes.

Similarly, every input layer node is connected to around 20% of the middle layer nodes. This way, only a small number of the RNN nodes receives the information about one particular input, and it is further spread across the network through the internal connections. On contrary, the eight output nodes are connected to all the ones in the middle layer. The $\alpha(\cdot)$ function plays the role of the kernel that converts pulses from one node into a continuous signal, then, such signals are summed up for all the nodes in the middle layer.

As described in the Appendix (A.5), the majority of network parameters are chosen to be either constants, or generated from some distribution of known parameters. The only parameters that have to be tuned later are the weights of the output layer connections, and W_{mE} , the parameter that defines the mean and variance of the Gaussian distribution used to obtain the weights connections of the excitatory and inhibitory cells. This parameter, effectively tunes the strength of these weights, and it will be chosen through the cross-validation.

Instead of tuning the hidden layer parameters through some iterative procedure, we will generate a random network in such a way to improve the chances for good outcome without doing any training. The output coefficients are set in the training phase, together with the value for W_{mE} . In the test phase, the eight outputs are calculated for the new set of inputs and the old coefficients, and the number of correct outputs will be counted. The 'correct' outputs are determined according to the criteria that will be described in what follows.

This procedure can be summarized in four steps:

- Initialize the network by choosing all the parameters of the input and hidden layer, and the time constants of the output layer. Select the parameter W_{mE} as well, in a systematic way, in order to obtain the best choice for the parameter in a series of tests.
- Calculate the remaining output coefficients by solving the linear regression, the procedure that will be described in the next section.
- Calculate the network outputs for the new set of inputs.
- Evaluate the mismatch between each output signal and the target signal. According to the errors obtained for the eight outputs, and the criteria that determines the allowed number of errors, select a group of 100 representative networks to be used in the classification tests.

5.5.2 Training phase

Writing the equation (5.13) in the discrete time, we obtain the following expression, also written in the vector form

$$\begin{aligned} u_i^{out}(t_n) &= \sum_{j,f} w_{ij}^{out} \alpha_{ij}(t_n - t_j^f) \\ &= \left[\sum_f \alpha_{i1}(t_n - t_1^f) \dots \sum_f \alpha_{iN}(t_n - t_N^f) \right] \cdot \begin{bmatrix} w_{i1}^{out} \\ w_{i2}^{out} \\ \dots \\ w_{iN}^{out} \end{bmatrix}. \end{aligned}$$

Written for the last d samples, this equation becomes

$$\begin{bmatrix} u_i^{out}(t_n) \\ \dots \\ u_i^{out}(t_{n-d+1}) \end{bmatrix} = K_i \cdot \begin{bmatrix} w_{i1}^{out} \\ w_{i2}^{out} \\ \dots \\ w_{iN}^{out} \end{bmatrix}.$$

where the matrix K_i contains all the kernel functions evaluated for the spike trains obtained in the considered interval in time

$$K_i = \begin{bmatrix} \sum_{j,f} \alpha_{i1}(t_n - t_j^f) & \dots & \sum_{j,f} \alpha_{i1}(t_{n-d+1} - t_j^f) \\ \dots & \dots & \dots \\ \sum_{j,f} \alpha_{iN}(t_n - t_j^f) & \dots & \sum_{j,f} \alpha_{iN}(t_{n-d+1} - t_j^f) \end{bmatrix}$$

Denote the target output for the node i as $y_i(t)$. This corresponds to the signals marked with green on the Figure 5.1. Suppose we use the last d samples for the training. As already stated, the considered model is discretized, together with its input, output and target signals. The d last samples of the target signal can be represented as $[y_i(t_n) \ y_i(t_{n-1}) \ \dots \ y_i(t-d)]^T$. Then, the set of output coefficients can be calculated as

$$\begin{bmatrix} w_{i1}^{out} \\ w_{i2}^{out} \\ \dots \\ w_{iN}^{out} \end{bmatrix} = K_i^{-1} \cdot \begin{bmatrix} y_i(t_n) \\ y_i(t_{n-1}) \\ \dots \\ y_i(t_{n-d+1}) \end{bmatrix}$$

which gives the expression for the calculation of output weight coefficients.

Test phase: The coefficients obtained during training are used to estimate the model outputs for a new set of input signals. Typically the outputs of the networks are very irregular, and, in order to estimate weather they correspond to the expected 'pulses' or not, the energy of the output was calculated during the interval of time where the pulse was expected. The output was considered to show a pulse whenever that energy exceeded the threshold value. For a series of input 'tasks', the number of output 'pulses' was calculated and the number of mismatches, with respect to the target signal. If this mismatch does not exceed 10%, the output is considered to work correctly.

The entire test phase included generation of many random networks and testing the number of correct outputs for each of them. This task with the linear outputs is not the ultimate goal of this study. Instead the equivalent classification as the one presented for the experimental data is performed. The procedure described here was used for network selection only. The idea was to test all the randomly generated networks under somewhat different conditions than for the main classification task, and chose the best among them that way. Also, since the procedure required generating and testing many networks, it was suitable to work with a method faster than SVM. The set of 200 networks is obtained this way. The tested network size was 100 and 500 cells.

The remaining network parameter W_{mE} was systematically chosen in advance, and for each tested value the percentage of selected among all the generated networks was calculated. The results are summarized in the Table 5.1 in order to illustrate the efficiency of the method, and the total number of generated networks needed to obtain the selected 200.

W_{mE}	0.001	0.003	0.005	0.01	0.015	0.02
% of selected networks	4.5	25	49	29	21	7

Table 5.1: The percent of selected networks among all the generated ones, as a function of the average connection strength in the hidden layer, W_{mE} . Illustration of the efficiency of the procedure, and the number of networks that has to be generated in order to have the 200 selections.

5.6 Tasks classification

For the set of 200 networks, obtained using the procedure described in the previous section, the similar classification task was imposed as in the case of experimental data analysis. The goal was to test the influence of some parameters, particularly the number of neurons, on the classification precision. The influence of the window length was also assessed, but the

obtained results revealed certain problems of the model than will be discussed later. For repeating these tests, the output linear layer was removed, and the spiking activity from the middle recurrent network was available. No particular tuning of the system parameters was done in order to ensure that this activity corresponds to the target, or mimic the motor intention. Still, keeping the 'good' set of parameters selected in the previous step guarantees good performance here as well.

5.6.1 Binary classification

The last step in the model study aims to mimic the method applied for the analysis of the experimental data. The spike trains collected from the recurrent neural network are converted into spike rates and the vectors of rates obtained for the whole set or a portion of the set of simulated artificial neurons. The rate vectors are used for training and testing an SVM type classifier, as before. The increasing network activity during the combination of pulses indicating a 'movement' should be detected from the rates. This increasing activity is the equivalent of 'motor intention' in our model. As another parallel to the experimental data study, the two input pulses will be called 'epochs', since they are created to mimic the experimental epochs CT, and RTh-MTh.

In order to repeat the two modes of data analysis presented before, the two procedures for calculating rates are applied, as shown on the Figure 5.3. In the first of them (second row on the figure), the integration time starts immediately after the rising edge of the target pulse and lasts for a certain interval of time defined in advance. In some of the tests, only the first W milliseconds after the start of the target signal were used, similarly as in the analysis presented in [77, 92]. Also, the entire duration of the 'epoch' was used for calculating rates, which corresponds to our first data analysis method that classifies experimental epochs according to the presence and absence of motor intention. Alternatively, the rates are calculated using the window of fixed size and dividing each epoch duration into the maximal number of non-overlapping segments, which is similar to our second data analysis method. In other words, three options for the integration time are applied, two of them repeating the method used in the previous tests on the experimental data, and one that was presented in the literature.

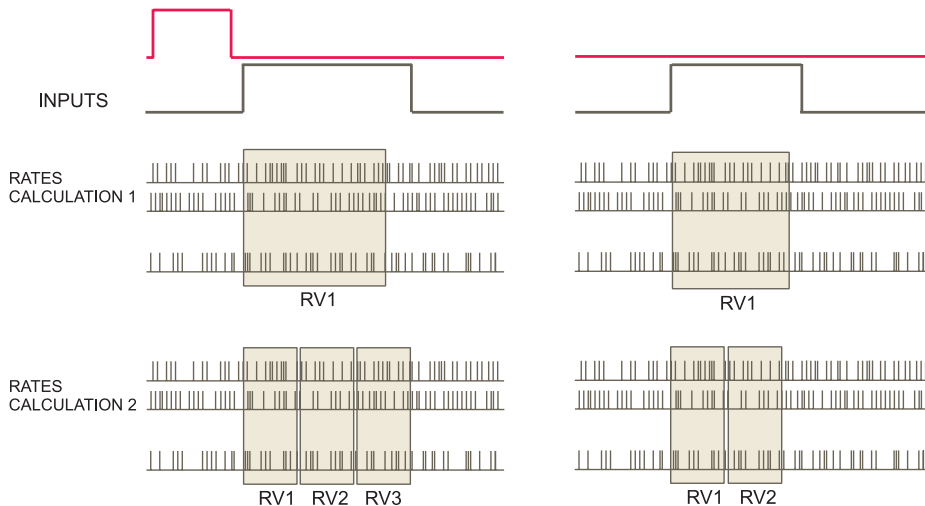


Figure 5.3: The two methods to calculate rates. Upper panels, the two types of input signals used in the model. Second row, the first method to calculate rates, corresponds to the epoch classification studied one the experimental data. Bottom row, the second method similar to the second method used for the experimental data.

The obtained rate vectors are used as the inputs to the classifier, as before. In the training phase, the first set of such data is used to tune the free parameters of the SVM algorithm, σ and C . Here, the simpler grid search method is applied along with the cross-validation. Then, the second set of data is generated and used to test the performance of the classifier.

The influence of two algorithm parameters is examined in this series of tests, the number of cells used for the classification, and the size of the window employed to calculate rates. Each of the performed tests can be summarized as follows,

- Select one of the generated 200 networks, each containing 100 cells,
- Chose the window size and the method for calculating spike rates; the tested values are $W \in \{10, 30, 50, 80\}$ milliseconds,
- Chose the percentage of cells in the set that will be used in the analysis, $M \in \{5, 10, 20, 30, 50\}\%$.
- Select the $M\%$ of all the available cells in the set, and for the selected set calculate the two groups of spike rates, for the training and for the test phase.
- Use the training set to obtain the optimal values for the parameters C and σ .
- For the optimal classifier obtained, use the test set of rate vectors to evaluate the method precision.
- Repeat the test with the same parameters M and W 100 times to obtain a statistically valid estimation of the classifier performance.

The method is applied for each of the 200 available networks. For each of them every value for M is tested. Also, all the listed methods for choosing W are implemented. For each combination (M, W) a random set of M cells is selected and the rates are calculated for each of them according to the given method and window length. Every test is repeated 100 times and the error estimation is obtained as the average over those values. Notice that the window can be 10, 30, 50, 80 milliseconds in these tests. The values for this parameter in the model are not comparable to the one in the experiments, since it scales with the network coefficients and is determined by the choice of the network parameters.

5.6.2 Results: Classification precision as a function of population size

In this section the results obtained from the described artificial network study will be presented. As already described, the 'motor intention' classification that mimic the analysis of the experimental data is carried. In fact, the system aims to identify the increase in the network activity resulting from the combination of input signals that signalize 'motion'. The goal of the study is not the sole classification of outputs, but examination of the influence of two parameters, the number of cells in the population used for information decoding, and the integration time allowed for rates calculation. Therefore, the two parameters were varied throughout the tests, the number of cells used to calculate inputs for the classifier, and the window size, W , used for the rates. The number of cells is represented as the percentage of the total number of cells in the set M , i.e. the portion of the network nodes. These parameters are tested for all combinations of values listed in the previous section, and using the three methods for rates calculation - positioning the window right after the start of the target pulse, using the entire duration of the target pulse (i.e. 'epoch'), and segmenting the pulse with the sliding window of fixed size, without overlapping.

The Figures 5.4 and 5.5 show the average performance of the classifier for certain values of M and W . For each pair of them, the 100 repetitions of the same trial is completed and

mean, maximal and minimal values are shown on the figure. The mean values lay on the curves shown on the figure, and the corresponding bars mark the maximal and the minimal value obtained. The only difference between the two figures is the protocol for using window when calculating rates. In the tests shown on the Figure 5.4 only one rate vector is evaluated for each trial, the window of fixed length is placed immediately after the start of the target signal. This way the window collects the 'first reaction' of the system to the rapid pulse input. In the second series of tests, window size is still fixed, but it samples the data from the entire interval corresponding to the target pulse. Therefore, some of the obtained rate vector will correspond to the beginning of the 'epoch', but the other will fall at its end.

Considering the window size influence, none of these tests correspond to the experimental evidences. There, the classification error constantly decreases as the window size increases, for all the tested windows. On the Figure 5.4, the similar result as in the experimental study is obtained for larger populations used, for at least 30% of the cells. For a smaller set of cells the error starts to increase after 50 milliseconds. For every population size, using the entire epoch instead of its fraction increases the error.

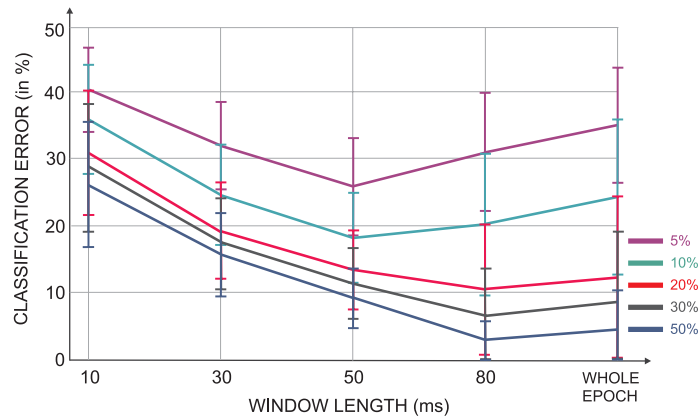


Figure 5.4: The average results obtained for the selected set of values for M and W . First method, only one rate vector per task calculated. The window is placed immediately after the start of the target pulse. The right most value - the result obtained when using the entire 'epoch' to calculate rates, added for comparison. Different curves correspond to the different number of cells used in the analysis. x axis - window size, y axis - the percent of correctly classified targets.

In the second test shown on the Figure 5.5, the error dependency on the window size is practically the opposite from expected. The error tends to increase with the window size, again being the biggest when the entire epoch is considered.

A possible explanation for these results is the network memory. As already stated, the implemented network possesses the short-term memory and its states (or nodes in this case) save the input information in certain form for some time. It is possible that in this case, the epoch duration exceeds the duration of network memory. While at the beginning, an intensive activity follows the signal appearance, at the end of the window it fades, making the end of the epoch very different with respect to the beginning. This can explain why are both methods sensitive to the window increase, and why using the entire epoch deteriorates the results. The comparison of the two methods shows that the second one outperforms the first one for the short windows, and becomes similar for the bigger windows. We may assume that this happen because the network needs certain time to integrate input information, therefore using only the small initial period of activity gives worst results than using many short windows

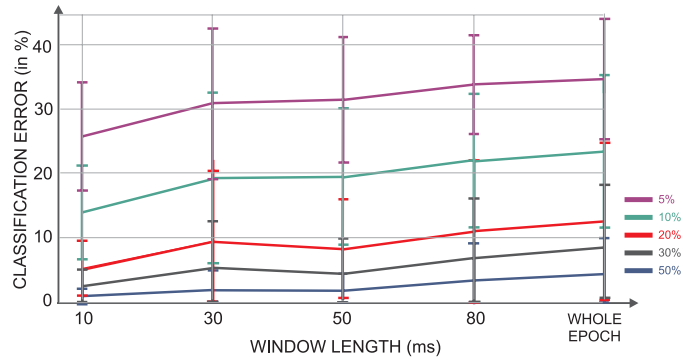


Figure 5.5: The average results obtained for the selected set of values for M and W . Several rate vectors per trial calculated, a window of fixed length is moved along the target pulse. The last value in both figures is the result obtained when using the entire 'epoch' to calculate rates, the result added for comparison. Different curves correspond to the different number of cells used in the analysis. x axis - window size, y axis - the percent of correctly classified targets.

sampling the entire epoch duration, even its ending. For a sufficiently large window, bigger than the half of the average epoch duration, both methods become the same.

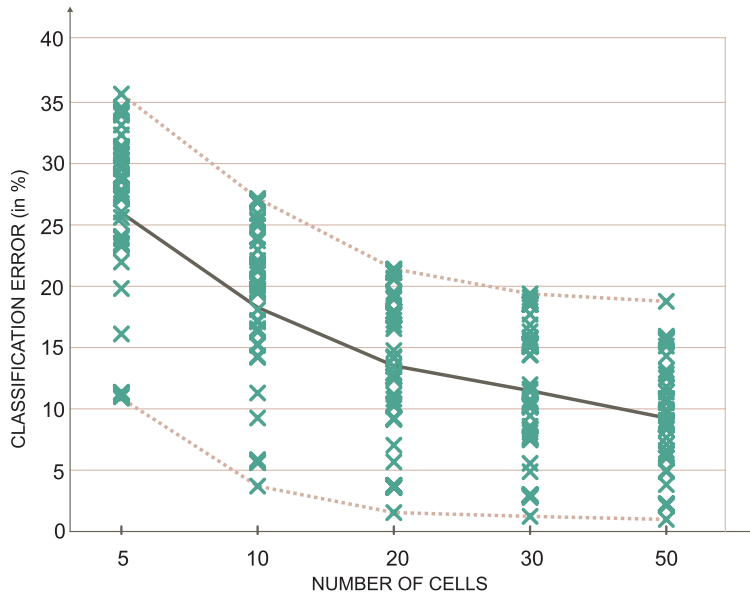


Figure 5.6: The classification error decreases for the increasing number of cells. Each green cross is the result obtained from one trial for the fixed window size of 30 milliseconds. For 5 cells the error is around 25%, but for 50 it falls below 10%. x axis - number of cells, y axis - classification error.

Although the study of the window length influence does not correspond entirely to our expectations, the error dependency on the population size does. The Figure 5.6 emphasizes this result. Here, the results are aligned according to the number of cells used in the analysis (note that since we use networks of 100 nodes, the percentage of cells M is equal to the number of cells). The green crosses represent the mean values obtained in a series of tests for the fixed window size of 30 milliseconds. Evidently, for the increasing population size the

error decreases. Similarly to our experimental results for 5 cells the error is around 25%, but for 50 cells it falls below 10%.

5.7 Conclusion

In this section we presented an artificial neural network, developed to simulate some selected characteristics of the experiments presented in the previous chapters. The input signals into the network mimic two tasks out of seven on the Figure 2.8, approximatively the RF and the NGO tasks. In the first of them, the initial period is defined by the central light on the screen. As long as it is on, the monkey should not move or plan a movement. Immediately after it goes off the target signal appears and the movement starts. In the model that should evoke an increase in the network activity. In the NGO task, after presentation of the control time and the target the monkey should stay immobile. In order to avoid using an additional instruction signal, which is given as a target color in the experiments, the absence of CT should indicate lack of movement. Of course, in the real experiments, it is necessary to signalize the start of the trial and prepare the monkey for what follows.

The central part of the model represents the recurrent network in the hidden layer. Such networks are proposed as the computational paradigm behind the short-term memory in the brains, and one of the principles for functioning of neural circuits [54, 53, 37]. Without discussing that assumption, we employed the RNN in this model in order to obtain a spiking network with the short-term memory. The spiking network was necessary to test the influence of the number of cells to the classification precision. Similarly, the memory was required for testing the window size effect on the performance, but also to maintain the minimum of characteristics of the original system studied.

As demonstrate in the previous section, the precision improvement due to the population size increase is justified here. This problem was also studied in the context of BCI literature. Several studies examined the minimal number of cells needed for the extraction of reliable control signals for the BCI [100, 77]. The presented model does not allow testing the window size influence reliably. The alternative systems, able to maintain longer memory, and even to solve the memory delay tasks will be studied in the future.

Chapter 6

Conclusion

The conclusion obtained from this study are summarized in this chapter. The possible future developments and additional tests are also listed here.

The central question in this study, the identification of motor intention in the neural signals, is formulated as a classification problem and solved using a traditional classification method. The first considered problem is the classification of experimental epochs, based on the average neural activity in each epoch calculated for a set of simultaneously recorded cells. The average obtained classification error falls between 25% and 35% of misclassification for the less successful database, and between 18% and 30% for the better one. Analyzing just the average performance is not sufficient in this context. The contribution of each epoch, and the error distribution per epoch has to be taken into account. The obtained results show good discrimination between the data collected in separate trials, but the identification of the change in motor intention during the same trial was a more difficult problem. It is expectable that various other signals influence the entire duration of a trial, and eventually corrupt the information of interest. Analysis of longer trials, with more than one alteration between movement and its absence can highlight this question. Similar test for a continuous movement could reveal interesting results. The systematically better results are obtained during the tasks and epochs related to the eye movement, and preparation for the eye movement. This might indicate that the eye-related signals stronger modulate the cell activity in this region. Generally, area 7 is considered to be a visually dominant region, but the study in [10] revealed the presence of hand-related as well as combinatorial cells along with the eye-dominant ones.

Throughout this study we included only the simultaneously recorded cells in the same decoding task. This resulted in small populations of up to 14 cells considered at the same time. This is also an argument that should be taken into account when evaluating the obtained classification results. It was possible to assess the influence of every single cell in so small populations. The obtained results reveal the three types of cells with respect to their contribution to motor intention coding. The majority of cells does not significantly influence the precision when removed from the set alone. Still, removing a critical number of such cells leads to degradation in performance. The other type are more important cells, which absence from the set significantly deteriorates the result. For only a few cells the improvement of the result was observed after removing the cell. The obtained results might illustrate a distributed and redundant coding of motor intention in the considered cortex region. Also, the fact that the majority of cells does not have a negative influence on the classification, might be a property of the cells alone, or the result of the robustness of the applied algorithm.

Finally, the dependency of the algorithm precision on the window length used to calculate spike rates is assessed through a separate analysis method. The obtained result is consistent with the reports in the literature [77]. The classification error decreases with the increasing

window size. This result cannot be directly compared to the first described study due to the difference in epoch treatment.

Several additional tests can be proposed in this context. A different segmentation of networks can be implemented, for example in such a way to compensate for the great variability in the epoch size. That would rule out a potential problem that could arise from this difference.

Also, an attempt to analyze the entire population of sequentially recorded cells can be made. Still a bigger number of available cells is not a guarantee of a successful result. It is not clear whether the motor intention manifests itself equally in the recordings collected during different sessions.

The second approach to the same problem is carried through the model study. As already described, the spiking recurrent network is employed together with the input-output protocol able to capture some relevant aspect of the experimental tasks. The influence of the two parameters was examined using the model, the considered population size and the length of the window used to calculate spike rates. The networks tested here contained 100 and 500 cells, the size imposed by the computational load. Still, it was sufficient to complete the analysis of interest. A subset of cells of a fixed size was selected randomly from the network, and the classification was done using only the spike trains corresponding to the selected cells. The examined populations were comparable or bigger than those used in the data analysis. The decrease of the classification error was evident for the increasing number of cells. This result indicates that the similar performance can be expected when using recordings from more cells in the task of motor intention classification from the experimental data.

For examining the influence of the window length, we needed a network of certain properties. First, a certain memory was required from the network, as well as in the choice of the input-output relation. Also, we implemented the spiking neuron model instead of the rate one, in order to have the same rate computation procedure as in the study of the experimental data.

The window size tests did not match entirely to our expectations. A probable reason behind it is the particular choice of network parameters that define too short network memory for the attempted task. In the context of the population size study, the model gave the expected behavior, indicating a probable precision improvement for the increasing number of cells.

The additional developments and tests that can be done for this model are numerous. First, a more challenging set of tasks can be formulated. It will be interesting to combine the inputs that mimic different sensory modalities, and observe the mechanisms of their combination in the network nodes, in order to produce a certain set of signals/actions at the output. Also, implementing the memory delay task can give a better framework for testing the window length impact. For example, a similar analysis as the one applied on the experimental data in [77] can be reproduced.

Many improvements related to the recurrent network itself can be made. The random generation of recurrent connections is the first and the simplest solution, and better methods for connectivity network generation can be proposed. An attempt to study the rejected, together with the accepted networks in the network selection procedure, described in the previous chapter, can give some indication what makes a network successful in solving the task of interest.

A more regular network topology can also be tested, for example a standard 3D column instead of a single layer of neurons. Also, the options for simulating somewhat bigger networks should be considered.

Appendix A

Spiking network: model and parameters tuning

A.1 Input signals

As described in the Section 5.3 the 'experimental tasks' are described through nine inputs, one for the control time/'go' signal, and eight for the eight targets. Each of them is represented as a pulse with unit amplitude and randomly chosen starting time T_s and duration T_d . Both of them are selected from a uniform distribution

$$T_s \sim [T_{s1}, T_{s2}], T_d \sim [T_{d1}, T_{d2}]$$

The duration of each 'task' (two inputs and a pause) is between 130 and 310 milliseconds. A series of such 'tasks' are concatenated to form the inputs in the training and in the test phase.

A.2 Input layer parameters

Each of the cells from the input layer is modeled as the standard LIF neuron, given as

$$\tau_i^{in} \frac{du_i(t)}{dt} = -u_i(t) + w_i^{in,e} R_i^{in} I_i^e(t), \quad \forall i \in 1, \dots, 9.$$

The resetting condition is defined as before, defining the minimal firing frequency of the spike trains transferred into the next layer

$$\text{if } u_i(t^*) = u_i^{th,in} \rightarrow u(t \in [t^*, t^* + \tau_i^{r,in}]) = u_i^{r,in}, \quad t^f = t^*.$$

The set of parameters describing the model are

- Membrane time constant for the input layer: τ_i^{in} ,
- Scaling coefficient for the input currents: $w_i^{in,e}$,
- Membrane resistance for the input layer cells R_i^{in} ,
- Input current pulse: $I_i^e(t)$,
- Refractory time for the input cells: $\tau_i^{r,in}$,
- Membrane potential in the resting state: $u_i^{r,in}$,
- Threshold potential: $u_i^{th,in}$.

A.3 Hidden layer / Recurrent neural network

The two types of cells are represented in this layer, the excitatory and the inhibitory cells, as usually in the neural networks. The same equations are used for both types of cells

$$\tau_i^x \frac{du_i(t)}{dt} = -u_i(t) + R_i^x [I_i^E(t) + I_i^I(t) + I_i^{in}(t) + I_i^{n,x}(t) + J_i^x], \quad x \in \{E, I\}.$$

The synaptic input currents from the excitatory and the inhibitory neurons are described as

$$\begin{aligned} I_i^x(t) &= \sum_{j,f} w_{ij}^x \alpha_{ij}(t - t_j^f) \\ &= \sum_{j,f} w_{ij}^x \cdot q^x \frac{t - \Delta_{ij}^x}{(\tau_{ij}^{s,x})^2} \cdot \exp\left(-\frac{t - \Delta_{ij}^x}{\tau_{ij}^{s,x}}\right) \cdot \Theta(t - \Delta_{ij}^x), \\ &\quad x \in \{E, I\}. \end{aligned}$$

The input currents from the first layer are given with

$$\begin{aligned} I_i^{in}(t) &= \sum_{f,j=1..9} w_{ij}^{in} \alpha_{ij}(t - t_j^f) \\ &= \sum_{f,j=1..9} w_{ij}^{in} \cdot q^{in} \frac{t - \Delta_{ij}^{in}}{(\tau_{ij}^{s,in})^2} \cdot \exp\left(-\frac{t - \Delta_{ij}^{in}}{\tau_{ij}^{s,in}}\right) \cdot \Theta(t - \Delta_{ij}^{in}). \end{aligned}$$

and the set of parameters is

- The membrane time constants for the excitatory and inhibitory cells in the RNN layer: τ_i^E, τ_i^I ,
- Membrane resistance for the both types of cells: R_i^E, R_i^I ,
- Internal weight connections in the RNN layer: w_{ij}^E, w_{ij}^I ,
- Rise and decay time constants for the post synaptic potential: $\tau_i^{s,E}, \tau_i^{s,I}$,
- Synaptic delay: $\Delta_{ij}^E, \Delta_{ij}^I$.
- Weights for the signals coming from the first layer: w_{ij}^{in} ,
- Rise and decay time constant for the inputs from the first layer: $\tau_i^{s,in}$,
- Synaptic delay for the signals from the first payer: Δ_i^{in} ,
- Noise for the excitatory and inhibitory cells: $I_i^{n,E}(t), I_i^{n,I}(t)$,
- Intrinsic cell current that enables cell spiking even in the absence of external inputs: J_i^E, J_i^I ,
- Refractory times: $\tau_i^{r,E}, \tau_i^{r,I}$,
- Resting potentials: $u_i^{r,E}, u_i^{r,I}$,
- Threshold potentials: $u_i^{th,E}, u_i^{th,I}$

A.4 Linear output

As already described in one of the previous sections, the outputs of the model, the linear combination of output spike trains is given as

$$u_i^{out} = R_i^{out} \sum_{j,f} w_{ij}^{out} \alpha_{ij}^{out}(t - t_j^f), \quad \forall i = 1..8,$$

$$\alpha_{ij}^{out}(t) = q^{out} \frac{t - \Delta_{ij}^{out}}{(\tau_{ij}^{s,out})^2} e^{-\frac{t - \Delta_{ij}^{out}}{\tau_{ij}^{s,out}}} \Theta(t - \Delta_{ij}^{out}),$$

where the model parameters are

- Output resistance: R_i^{out} ,
- Synaptic time constants $\tau_{ij}^{s,out}$ and synaptic delay Δ_{ij}^{out} .

the value q^{out} , as well as the equivalent parameters in the other layers is not included among the network parameters, since it is calculated from the requirement that the maximum of the function $\alpha(\cdot)$ should be 1.

A.5 Parameters selection

The hidden layer parameters are chosen according to the general recipes for generating a RNN, and the values proposed in [14]. The publicly available software package used for some of the tests here also proposes some default values.

Input layer

- The membrane time constant: $\tau_i^{in} = 30ms$.
- Scaling coefficients for the input currents are equal for each cell: $w_i^{in} = 0.01$. Each signal from the Figure 5.1 is input to one cell from this layer, and the output spike trains are transferred to multiple cells in the RNN.
- Membrane resistance: $R_i^{in} = 10 M\Omega$.
- Refractory times for the input cells correspond to those of the excitatory neurons: $\tau_i^{r,in} = 3ms$.
- The reset potential is chosen from the uniform distribution: $u_i^{r,in} \sim \text{Unif}[13.5, 14.5] mV$.
- The threshold potential: $u_i^{th,in} = 15mV$.

Recurrent network parameters

- The membrane time constants are chosen to be equal for all cells: $\tau_i^E = \tau_i^I = 30ms$.
- The membrane resistances are also equal: $R_i^E = R_i^I = 10M\Omega$.
- Synaptic time constants for excitatory and inhibitory neurons are chosen from the uniform distribution, but with different distribution parameters:

- The connection strength for the excitatory synapses is randomly generated from the Gaussian distribution: $w_{ij}^E \sim \text{Gauss}(W_{mE}, 10^{-3}W_{mE})$,
- The connection weight for the inhibitory synapses is also generated from the Gaussian distribution as $w_{ij}^I \sim \max \text{Gauss}(-2E_{mE}, 10^{-3}W_{mE})$,
- The synapse time constants are chosen from the uniform distributions from the different intervals for the excitatory and the inhibitory cells
 $\tau_{ij}^{s,E} \sim \text{Unif}[2.5, 3.5] \text{ ms}$, $\tau_{ij}^{s,I} \sim \text{Unif}[5, 7] \text{ ms}$.
- Synaptic delay is derived from the same uniform distribution for both types of cells:
 $\Delta_{ij}^E, \Delta_{ij}^I \sim \text{Unif}[0.1, 1]$
- Weights for the connections coming from the first layer: $w_{ij}^{in} = 0.01$,
- Time constants for the connections from the input layer: $\tau_{ij}^{s,in} \sim \text{Unif}[2.5, 3.5]$,
- The synaptic transmission delay from the input layer: $\Delta_{ij}^{in} \sim \text{Unif}[0.1, 1]$,
- The Gaussian noise is included in every cell membrane equation, and is equal for the excitatory and the inhibitory cells: $I_i^{n,E}, I_i^{n,I} \sim \text{Gauss}(0, \sigma_i^n)$, where $\sigma_i^n = 5mV$.
- The auto inject current, the parameter that enables cell spiking with certain frequency in the absence of external currents, is also chosen from the uniform distribution: $J_i^E, J_i^I \sim \text{Unif}[13.5, 14.5] \text{ nA}$,
- The refractory times are fixed for both types of cells: $\tau_i^{r,E} = 3ms$, $\tau_i^{r,I} = 2ms$,
- The resting potentials are chosen the same way as for the input layer cells: $u_i^{r,E}, u_i^{r,I} \sim \text{Unif}[13.5, 14.5] \text{ mV}$,
- Finally, the threshold potentials are set as: $u_i^{th,E} = u_i^{r,I} = 15mV$.

Output layer

- The output resistance R_i^{out} is not chosen separately, since it can be taken into account with the weight coefficient w_{ij}^{out} . This parameter is chosen in the training procedure, described in the Section 5.5.
- The synapse time constant is set as: $\tau_{ij}^{s,out} = 20ms$,
- and there is no need to define the output transmission delay since it has no influence on the network behavior: $\Delta_{ij}^{out} = 0$.

Bibliography

- [1] CSIM: A neural Circuits SIMulator. Software available at the web page of the group: <http://www.lsm.tugraz.at/csim/index.html>. [cited at p. 94]
- [2] N. Achtman, A. Afshar, G. Santhanam, B. M. Yu, S. I. Ryu, and K. V. Shenoy. Free-paced high-performance brain-computer interfaces. *J. Neural Eng.*, 4:336–347, 2007. [cited at p. 44, 45, 46, 47, 49, 53, 75, 82, 83, 84]
- [3] B. Z. Allison, E. Winter Wolpaw, and J. R. Wolpaw. Brain-computer interface systems: progress and prospects. *Expert Rev. Med. Devices*, 4:463–474, 2007. [cited at p. 38, 39, 40]
- [4] R. A. Andersen and C. A. Buneo. Intentional maps in posterior parietal cortex. *Annu. Rev. Neurosci.*, 25:189–220, 2002. [cited at p. 1, 12, 14, 16, 17, 22, 27, 28, 30, 33, 34, 49]
- [5] R. A. Andersen, L. H. Snyder, D. C. Bradley, and J. Xing. Multimodal representation of space in the posterior parietal cortex and its use in planning movements. *Annu. Rev. Neurosci.*, 20:303–330, 1997. [cited at p. 13, 28]
- [6] S. N. Backer, N. Philbin, R. Spinks, E. M. Pinches, D. M. Wolpert, D. G. MacManus, Q. Pauluis, and R. N. Lemon. Multiple single unit recording in the cortex of monkeys using independently movable microelectrodes. *Journal of Neuroscience Methods*, 94:5–17, 1999. [cited at p. 17]
- [7] A. Battaglia-Mayer, P. S. Archambault, and R. Caminiti. The cortical network for eye-hand coordination and its relevance to understanding motor disorders of parietal patients. *Neuropsychologia*, 44:2607, 2006. [cited at p. 5, 6, 16, 27, 37]
- [8] A. Battaglia-Mayer, S. Ferraina, T. Mitsuda, B. Marconi, A. Genovesio, P. Onorati, F. Lacquaniti, and R. Caminiti. Early coding of reaching in the parietooccipital cortex. *J. Neurophysiology*, 83:2374–2391, 2000. [cited at p. 16, 27]
- [9] A. Battaglia-Mayer, M. Mascaro, E. Brunamonti, and R. Caminiti. The over-representation of contralateral space in parietal cortex: A positive image of directional motor components of neglect? *Cerebral Cortex*, 15:514–525, May 2005. [cited at p. 1, 2, 11, 12, 13, 16, 17, 20, 24, 27, 34, 67, 68]
- [10] A. Battaglia-Mayer, M. Mascaro, and R. Caminiti. Temporal evolution and strength of neural activity in parietal cortex during eye and hand movement. *Cerebral Cortex*, 17:1350–1363, June 2007. [cited at p. 1, 2, 11, 17, 20, 24, 34, 47, 67, 68, 103]
- [11] M. J. Black, E. Bienenstock, J. P. Donoghue, M. Serruya, W. Wu, and Y. Gao. Connecting brains with machines: The neural control of 2d cursor movement. *In Proceedings 1st int. IEEE EMBS Conf. Neural Eng. Capri Island, Italy*, 2003. [cited at p. 13, 45]

- [12] C. A. Buneo and R. A. Andersen. The posterior parietal cortex: Sensorimotor interface for the planning and online control of visually guided movements. *Neuropsychologia*, 44:2594–2606, 2006. [cited at p. 11, 16, 27]
- [13] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998. [cited at p. 57]
- [14] H. Burgsteiner, M. Kröll, A. Leopold, and G. Steinbauer. Movement prediction from real-world images using a liquid state machine. *Applied Intelligence*, 26:99–109(11), 2007. [cited at p. 94, 107]
- [15] R. Caminiti, S. Ferraina, and P. B. Johnson. The sources of visual information to the primate frontal lobe: A novel role for the superior parietal lobule. *Cerebral Cortex*, 6:319–328, 1996. [cited at p. 10, 11]
- [16] J. M. Carmena, M. A. Lebedev, R. E. Crist, E. O’Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis. Learning to control a brain-machine interface for reaching and grasping by primates. *PLoS Biology*, 1:193–207, 2003. [cited at p. 1, 13, 30, 34, 37, 40, 43, 44, 49, 75, 84]
- [17] J. M. Carmena, M. A. Lebedev, C. S. Henriquez, and M. A. L. Nicolelis. Stable ensemble performance with single-neuron variability during reaching movements in primates. *The Journal of Neuroscience*, 25:10712–10716, 2005. [cited at p. 13, 40, 43, 49]
- [18] C. C. Chang and C. J. Lin. Libsvm: a library for support vector machines. software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>. 2001. [cited at p. 56, 57, 61, 76]
- [19] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing multiple parameters for support vector machines. *Machine Learning*, 46:131–159, 2002. [cited at p. 62]
- [20] J. K. Chapin, K. A. Moxon, R. S. Markowitz, and M. A. L. Nicolelis. Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nature neuroscience*, 2:664–670, 1999. [cited at p. 13, 43]
- [21] Y. E. Cohen and R. A. Andersen. A common reference frame for movement plans in the posterior parietal cortex. *Nature Reviews Neuroscience*, 3:553–562, 2002. [cited at p. 10, 11, 12, 13, 14, 15]
- [22] C. L. Colby and M. E. Goldberg. Space and attention in parietal cortex. *Annu. Rev. Neurosci.*, 22:319–349, 1999. [cited at p. 10, 11, 12, 13]
- [23] A. Compte, N. Brunel, P. S. Goldman-Rakic, and X.-J. Wang. Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model. *Cerebral Cortex*, 10:910–923, September 2000. [cited at p. 87]
- [24] P. Dayan and L. F. Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. The MIT Press, 2001. [cited at p. 87]
- [25] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, second edition, 2001. [cited at p. 57, 63]
- [26] R. Eckhorn and U. Thomas. A new method for the insertion of multiple microprobes into neural and muscular tissue, including fiber electrodes, fine wires, needles and micro-sensors. *Journal of Neuroscience Methods*, 49:175–179, 1993. [cited at p. 17]

- [27] L. A. Farwell and D. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology*, 70:510–523, 1988. [cited at p. 39]
- [28] E. A. Felton, J. A. Wilson, J. C. Williams, and P. C. Garell. Electrocorticographically controlled brain-computer interfaces using motor and sensory imagery in patients with temporary subdural electrode implants. *J. Neurosurg.*, 106:495–500, 2007. [cited at p. 39]
- [29] S. Ferraina, P.B. Johnson, M.R. Garasto, A. Battaglia-Mayer, L. Ercolani, L. Bianchi, F. Lacquaniti, and R. Caminiti. Combination of hand and gaze signals during reaching: Activity in parietal area 7m of the monkey. *J. Neurophysiol.*, 77:1034–1038, 1997. [cited at p. 11, 12]
- [30] M. Flanders, S.I. Helms-Tillery, and J. F. Soechting. Early stages in a sensorimotor transformation. *Behavior and brain sciences*, 15:309–362, 1992. [cited at p. 16, 27]
- [31] F. Friedrichs and C. Igel. Evolutionary tuning of multiple svm parameters. *Neurocomputing*, 64:107–117, 2005. [cited at p. 62]
- [32] A. P. Georgopoulos. Higher order motor control. *Annu. Rev. Neurosci.*, 14:361–377, 1991. [cited at p. 12, 13, 42]
- [33] A. P. Georgopoulos, J. F. Kalaska, R. Caminiti, and J. T. Massey. On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *The Journal of Neuroscience*, 2:1527–1537, 1982. [cited at p. 12, 37]
- [34] A. P. Georgopoulos, A. B. Schwartz, and R. E. Kettner. Neuronal population coding of movement direction. *Science*, 233:1416–1419, 1986. [cited at p. 12, 13, 37, 42]
- [35] W. Gerstner and W. M. Kistler. *Spiking Neuron Models*. Cambridge University Press, August 2002. [cited at p. 87, 88]
- [36] The IGI LSM Group. *CSIM: A Neural Circuit SIMulator*. [cited at p. 94]
- [37] S. Haeusler and W. Maass. A statistical analysis of information processing properties of lamina-specific cortical microcircuit models. *Cerebral Cortex*, 17:149–162, 2007. [cited at p. 86, 101]
- [38] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. A. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn, and J. P. Donoghue. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442:164–171, 2006. [cited at p. 45]
- [39] R. E. Isaacs, D. J. Weber, and A. B. Schwartz. Work toward real-time control of a cortical neural prosthesis. *IEEE Transactions on Rehabilitation Engineering*, 8:196–198, 2000. [cited at p. 37, 42, 84]
- [40] H. Jaeger. *A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach*. Fraunhofer Institute for Autonomous Intelligent Systems (AIS). International University Bremen, March 2005. [cited at p. 86]
- [41] P. B. Johnson, S. Ferraina, L. Bianchi, and R. Caminiti. Cortical networks for visual reaching: Physiological and anatomical organization of frontal and parietal lobe arm regions. *Cerebral Cortex*, 6:102–119, 1996. [cited at p. 37]
- [42] P. Joshi and W. Maass. Movement generation with circuits of spiking neurons. *Neural Computation*, 17:1715–1738, 2005. [cited at p. 86]

- [43] J. F. Kalaska, R. Caminiti, and A. P. Georgopoulos. Cortical mechanisms related to the direction of two-dimensional arm movements: Relations in parietal area 5 and comparison with motor cortex. *Exp. Brain Res.*, 51:247–260, 1983. [cited at p. 12]
- [44] E. R. Kandel, J. H. Schwartz, and T. M. Jessell. *Principles of Neural Science*. McGraw-Hill Companies, 2000. [cited at p. 3, 4]
- [45] M. Kawato. Brain controlled robots. *HFSP Journal*, 2:136–142, 2008. [cited at p. 43]
- [46] P. Kennedy, D. Andreasen, P. Ehirim, B. King, T. Kirby, H. Mao, and M. Moore. Using human extra-cortical local field potentials to control a switch. *J. Neural Eng.*, 1:72–77, 2004. [cited at p. 46]
- [47] P. R. Kennedy, R. A. E. Bakay, M. M. Moore, K. Adams, and J. Goldwaihthe. Direct control of a computer from the human central nervous system. *IEEE Trans. Rehabil. Eng.*, 8:198–202, 2000. [cited at p. 41, 45]
- [48] S. Kirkpatrick. Optimization by simulated annealing: Quantitative studies. *Journal Stat. Physics*, 34:975–986, 1984. [cited at p. 62]
- [49] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983. [cited at p. 62]
- [50] M. A. Lebedev and M. A. L. Nicolelis. Brain-machine interfaces: past, present and future. *Trends in Neurosciences*, 29:536–546, 2006. [cited at p. 34, 39]
- [51] E. C. Leuthard, G. Schalk, J. R. Wolpaw, J. G. Ojemann, and D. W. Moran. A brain-computer interface using electrocorticographic signals in humans. *J. Neural Eng.*, 1:63–71, 2004. [cited at p. 39]
- [52] J. W. Lewis and D. C. Van Essen. Mapping of architectonic subdivisions in the macaque monkey, with emphasis on parieto-occipital cortex. *The Journal of Comparative Neurology*, 428:79–111, 2000. [cited at p. 7]
- [53] W. Maass, T. Natschläger, and H. Markram. *A model for real-time computation in generic neural microcircuits*, volume 15. MIT Press, Cambridge, MA. [cited at p. 86, 101]
- [54] W. Maass, T. Natschläger, and H. Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14:2531–2560, 2002. [cited at p. 86, 101]
- [55] H. Markram, Y. Wang, and M. Tsodyks. Differential signaling via the same axon of neocortical pyramidal neurons. *Proc. Natl. Acad. Sci.*, 95:5323–5328, 1998. [cited at p. 89]
- [56] J. Kent Martin and D. S. Hirschberg. Small sample statistics for classification error rates i: Error rate measurements. Technical report, University of California, Irvine, 1996. Technical Report No. 96-21. [cited at p. 52, 53]
- [57] J. Kent Martin and D. S. Hirschberg. Small sample statistics for classification error rates ii: Confidence intervals and significance tests. Technical report, University of California, Irvine, 1996. Technical Report No. 96-21. [cited at p. 52, 53]
- [58] S. G. Mason, A. Bashashati, M. Fatourech, K. F. Navarro, and G. E. Birch. A comprehensive survey of brain interface technology designs. *Annals of Biomedical Engineering*, 35(2):137–169, 2007. [cited at p. 38, 39]

- [59] D. J. McFarland, D. J. Krusienski, W. A. Sarnacki, and J. R. Wolpaw. Emulation of computer mouse control with a noninvasive brain-computer interface. *Journal of Neural Engineering*, 5:101–11, 2008. [cited at p. 39]
- [60] J. McIntyre, F. Stratta, and F. Laccquaniti. Short-term memory for reaching to visual targets: Psychophysical evidence for body-centered reference frames. *The Journal of Neuroscience*, 18(20):8423–8435, 1998. [cited at p. 16, 27]
- [61] V. Mountcastle. The evolution of ideas concerning the function of the neocortex. *Cerebral Cortex*, 5:289–295, 1995. [cited at p. 4]
- [62] V. B. Mountcastle, J. C. Lynch, A. Georgopoulos, H. Sakata, and C. Acuna. Posterior parietal association cortex of the monkey: Command functions for operations within extrapersonal space. *Journal of Neurophysiology*, 38:871–908, 1975. [cited at p. 5, 11, 12, 13]
- [63] K. R. Müller, M. Krauledat, G. Dornhege, G. Curio, and B. Blankertz. Machine learning techniques for brain-computer interfaces. *Biomed. Tech*, 49:11–22, 2004. [cited at p. 52, 53]
- [64] A. Murata, V. Gallese, M. Kaseda, and H. Sakata. Parietal neurons related to memory-guided hand manipulation. *Journal of Neurophysiology*, 75:2180–2186, May 1996. [cited at p. 11]
- [65] S. Musallam, B. D. Corneil, B. Greger, H. Scherberger, and R. A. Anderson. Cognitive control signals for neural prosthetics. *Science*, 305:258–262, 2004. [cited at p. 13, 37, 44, 45, 46, 47, 49, 53, 71, 75, 82, 84]
- [66] T. Natschläger and W. Maass. Finding the key to a synapse. Technical report, Institute for Theoretical Computer Science, Technische Universität Graz, Austria. [cited at p. 89]
- [67] T. Natschläger, H. Markram, and W. Maass. *Neuroscience databases. A practical guide*, chapter Computer models and analysis tools for neural microcircuits. R. Kötter (Ed.), (pp. 123–138). Boston: Kluwer Academic Publishers. [cited at p. 89]
- [68] M. A. L. Nicolelis, D. Dimitrov, J. M. Carmena, R. Crist, G. Lehew, J. D. Kralik, and S. P. Wise. Chronic, multisite, multielectrode recordings in macaque monkeys. *PNAS*, 100:11041–11046, 2003. [cited at p. 40, 41, 43, 49]
- [69] B. P. Olson, J. Si, J. Hu, and J. He. Closed-loop cortical control of direction using support vector machines. *IEEE Trans. Neural Sys. Rehab. Eng.*, 13:72–80, 2005. [cited at p. 46]
- [70] Y. Y. Ou, C. Y. Yhen, S. C. Hwang, and Y. J. Oyang. Expediting model selection for support vector machines based on data reduction. *In IEEE International Conference on Systems, Man and Cybernetics*, 2003. [cited at p. 56]
- [71] L. Paninski, M. R. Fellows, N. G. Hatsopoulos, and J. P. Donoghue. Spatiotemporal tuning of motor cortical neurons for hand position and velocity. *J. Neurophysiol.*, 91:515–532, 2004. [cited at p. 45, 53]
- [72] B. Pesaran, J. S. Pezaris, M. Sahani, P. P. Mitra, and R. A. Andersen. Temporal structure in neuronal activity during working memory in macaque parietal cortex. *Nature Neuroscience*, 5:805–811, august 2002. [cited at p. 13]
- [73] E. A. Pohlmeyer, S. A. Solla, E. J. Perreault, and L. E. Miller. Prediction of upper limb muscle activity from motor cortical discharge during reaching. *J. Neural Eng.*, 4:369–379, 2007. [cited at p. 46]

- [74] H. Sakata, M. Taira, A. Murata, and S. Mine. Neural mechanisms of visual guidance of hand action in the parietal cortex of the monkey. *Cerebral Cortex*, 5:429–438, 1995. [cited at p. 11]
- [75] E. Salinas and L. F. Abbott. A model of multiplicative neural responses in parietal cortex. *Proc. Natl. Acad. Sci. USA*. [cited at p. 2, 14, 15, 86]
- [76] J. C. Sanchez, J. M. Carmena, M. A. Lebedev, M. A. L. Nicolelis, J. G. Harris, and J. C. Principe. Ascertaining the importance of neurons to develop better brain-machine interfaces. *IEEE Transactions on Biomedical Engineering*, 51:943–953, 2004. [cited at p. 43, 71]
- [77] G. Santhanam, S. I. Ryu, B. M. Yu, A. Afshar, and K. V. Shenoy. A high-performance brain-computer interface. *Nature*, 442:195–198, 2006. [cited at p. 1, 13, 30, 37, 45, 53, 74, 83, 97, 101, 103, 104]
- [78] H. Scherberger and R. A. Andersen. Target selection signals for arm reaching in the posterior parietal cortex. *The Journal of Neuroscience*, 27:2001–2012, 2007. [cited at p. 49, 84]
- [79] H. Scherberger, M. R. Jarvis, and R. A. Andersen. Cortical local field potential encodes movement intentions in the posterior parietal cortex. *Neuron*, 46:347–354, 2005. [cited at p. 37, 46, 47, 48, 82, 83]
- [80] A. B. Schwartz. Direct cortical representation of drawing. *Science*, 265:540–542, 1994. [cited at p. 42, 49]
- [81] A. B. Schwartz. Cortical neural prosthetics. *Annu. Rev. Neurosci.*, 27:487–507, 2004. [cited at p. 40, 41]
- [82] A. B. Schwartz, D. W. Moran, and A. Reina. Differential representation of perception and action in the frontal cortex. *Science*, 303:380–383, 2004. [cited at p. 42]
- [83] M. D. Serruya, N. G. Hastopoulos, L. Paninski, M. R. Fellows, and J. P. Donoghue. Instant neural control of a movement signal. *Nature*, 416:141–142, 2002. [cited at p. 1, 13, 30, 34, 37, 44, 45, 84]
- [84] K. V. Shenoy, D. Meeker, S. Cao, S. A. Kureshi, B. Pesaran, C. A. Buneo, A. P. Batista, P. P. Mitra, J. W. Burdick, and R. A. Andersen. Neural prosthetic control signals from plain activity. *NeuroReport*, 14:591–596, March 2003. [cited at p. 13, 30, 37, 44, 45, 46, 47, 49, 82, 83, 84]
- [85] D. J. Sheskin. *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, CRC Press LLC, 2000. [cited at p. 54]
- [86] L. H. Snyder, A. P. Batista, and R. A. Andersen. Coding of intention in the posterior parietal cortex. *Nature*, 386:167–170, 1997. [cited at p. 33, 47]
- [87] L. H. Snyder, A. P. Batista, and R. A. Andersen. Intention-related activity in the posterior parietal cortex: a review. *Vision Research*, 40:1433–1441, 2000. [cited at p. 10, 33]
- [88] C. Staelin. Parameter selection for support vector machines. *Technical Report HPL-2002-354, HP Laboratories, Israel*, 2003. [cited at p. 62]

- [89] S. Suner, M. R. Fellows, C. Vargas-Irwin, G. K. Nakata, and J. P. Donoghue. Reliability of signals from a chronically implanted silicon-based electrode array in non-human primate primary motor cortex. *IEEE Trans. on Neural Syst. and Rehab. Eng.*, 13:524–541, 2005. [cited at p. 40, 41]
- [90] J. S. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002. [cited at p. 57, 61]
- [91] D. M. Taylor, S. I. Helms Tillery, and A. B. Schwartz. Direct cortical control of 3d neuroprosthetic devices. *Science*, 296:1829–1832, 2002. [cited at p. 37, 42]
- [92] D. M. Taylor, S. I. Helms Tillery, and A. B. Schwartz. Information conveyed through brain-control: Cursor versus robot. *IEEE Trans. Neural Sys. Rehab. Eng.*, 11:195–199, 2003. [cited at p. 13, 42, 44, 97]
- [93] S. I. Helms Tillery, D. M. Taylor, and A. B. Schwartz. The general utility of a neuroprosthetic device under direct cortical control. *Proceedings of the 25th Annual International Conference of the IEEE EMBS*, pages 2043–2046, 2003. [cited at p. 42]
- [94] T. P. Trappenberg. *Fundamentals of Computational Neuroscience*. Oxford University Press, 2002. [cited at p. 4]
- [95] W. Truccolo, G. M. Friehs, J. P. Donoghue, and L. R. Hochberg. Rprimary motor cortex tuning to intended movement kinematics in humans with tetraplegia. *The Journal of Neuroscience*, 28:1163–1178, 2008. [cited at p. 45]
- [96] M. Tsodyks and H. Markram. The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability. *Proc. Natl. Acad. Sci. USA*, 94:719–723, 1997. [cited at p. 89]
- [97] J. Vandekerckhove, 2006. The Matlab implementation of the simulated annealing algorithm. Downloadable from MathWorks. [cited at p. 62]
- [98] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz. Cortical control of a prosthetic arm for self-feeding. *Nature*, 453:1098–1101, 2008. [cited at p. 42]
- [99] M. Weinrich and S. P. Wise. The premotor cortex of the monkey. *The Journal of Neuroscience*, 2(9), 1982. [cited at p. 12]
- [100] J. Wessberg, C. R. Stambaugh, J. D. Kralik, P. D. Beck, M. Laubach, J. K. Chapin, J. Kim, S. H. Biggs, M. A. Srinivasan, and M. A. L. Nicolelis. Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature*, 408:361–365, 2000. [cited at p. 40, 43, 44, 53, 71, 75, 84, 101]
- [101] S. P. Wise, D. Boussaoud, P. B. Johnson, and R. Caminiti. Premotor and parietal cortex: Corticocortical connectivity and combinatorial computations. *Annu. Rev. Neurosci.*, 20:25–42, 1997. [cited at p. 10, 12]
- [102] J. Xing and R. A. Andersen. Models of the posterior parietal cortex which perform multimodal integration and represent space in several coordinate frames. *Journal of Cognitive Neuroscience*, 12(4):601–614, 2000. [cited at p. 2, 15, 87]
- [103] T. Zhang, H. W. Heuer, and K. H. Britten. Parietal area vip neuronal responses to heading stimuli are encoded in head-centered coordiantes. *Neuron*, 42:993–1001, June 2004. [cited at p. 11]

A.6 CV

CURRICULUM VITAE

Name: Jugoslava Aćimović.
Address: Rue Marterey 34, 1005 Lausanne, Switzerland.

Phone: (mob.) +41 76 53.93.423
(home.) +41 21 63.55.404
e-mail: jugoslava.acimovic@epfl.ch, jugoslava.acimovic@gmail.com.

Born: 6th July 1978, Valjevo, Serbia.
Nationality: Serbian.

Education:

- Postgraduate studies - École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland.
 - November 2004 - present: PhD studies at EPFL, Laboratory of Nonlinear Systems (LANOS). Interested in problems related to signal processing, nonlinear systems theory, machine learning, time series analysis, bio-inspired modeling and data analysis, complex networks.
 - April-October 2008: Visiting Tampere University of Technology, Computational Systems Biology Group.
 - July-September 2006: Visiting Katolieke Universiteit Leuven, group for Signals, Identification, Systems Theory and Automation (SISTA).
 - J. Acimovic, A. Battaglia-Mayer, R. Caminimi, M. Hasler. “Machine-learning based identification of motor intention in the parietal cortex”. *Poster session at the 1st INCF Congress of Neuroinformatics (2008)*, Stockholm, Sweden.
 - August 2006: Workshop on Advances in Information Technology (Bertinoro, Italy). Presented a poster entitled: “Connectivity Patterns in Small Neuronal Ensembles”.
 - September 2005: Participate in summer school on Synchronization in Nonlinear Systems and Complex Networks (Florence, Italy).
 - October 2003 - July 2004: Predoctoral program at EPFL. Completed courses and project related to signal processing, wavelets theory, communication theory and sensor networks.
- Undergraduate studies - Faculty of Electrical Engineering, University of Belgrade, Serbia.
 - October 1997 -July 2003: Undergraduate studies. Graduated with “Bachelor of Science in Electrical Engineering”, specialized in communicated systems and signal processing.
 - 1999-2003: Recipient of the scholarship of Serbian Ministry of Science and Technology.
 - October - November 2002: Visiting student at Escuela Universitaria Politecnica de Mataro, Spain.

Previous work experience:

- Teaching assistant for the courses “Circuits and Systems” and “Pattern Classification and Machine Learning”.
- Supervising two student projects related to machine learning methods for time series analysis.

Language proficiency: Serbian: mother language.
English: fluently.
French: fair.

Programming skills: C, Matlab.

Publications:

- J. Acimovic, R. Cristescu, B. Beferull-Lozano, “**Efficient Distributed Multiresolution Processing for Data Gathering in Sensor Networks**”. *International Conference on Acoustic, Speech, and Signal Processing (ICASSP 2005)*, Philadelphia, PA, USA.
- J. Acimovic, B. Beferull-Lozano, R. Cristescu, “**Adaptive Distributed Algorithm for Power-Efficient Data Gathering in Sensor Networks**”. In *Proceedings of the IEEE International Symposium on Wireless Sensor Networks (WirelessCom 2005)*. Maui, HI.
- J. Acimovic, A. Battaglia-Mayer, R. Caminiti, M. Hasler. “**Automatic Methods for Motor Intention Recognition from Spike Rates**”. *International Symposium on Nonlinear Theory and its Applications (NOLTA 2007)*, Vancouver, Canada.
- A. Palladini, J. Acimovic, M. Hasler. “**Modeling behavioral tasks to asses visual information integration and motor information encoding in parietal cortex**”. *International Symposium on Nonlinear Theory and its Applications (NOLTA 2008)*, Budapest, Hungary.
- J. Acimovic, A. Battaglia-Mayer, R. Caminiti, M. Hasler. “**Machine Learning Methods for Motor Intention Detection from Spike Train Recordings**”. To be submitted.