The Role of Simulation in Developing and Designing Applications for 2-Class Motor Imagery Brain-Computer Interfaces

Melissa Quek

Submitted in fulfilment
of the requirements for the
Degree of Doctor of Philosophy

School of Computing Science
College of Science and Engineering
University of Glasgow

July 2013

Abstract

A Brain-Computer Interface (BCI) can be used by people with severe physical disabilities such as Locked-in Syndrome (LiS) as a channel of input to a computer. The time-consuming nature of setting up and using a BCI, together with individual variation in performance and limited access to end users makes it difficult to employ techniques such as rapid prototyping and user centred design (UCD) in the design and development of applications. This thesis proposes a design process which incorporates the use of simulation tools and techniques to improve the speed and quality of designing BCI applications for the target user group.

Two different forms of simulation can be distinguished: *offline* simulation aims to make predictions about a user's performance in a given application interface given measures of their baseline control characteristics, while *online* simulation abstracts properties of interaction with a BCI system which can be shown to, or used by, a stakeholder in real time. Simulators that abstract properties of BCI control at different levels are useful for different purposes. Demonstrating the use of offline simulation, Chapter 3 investigates the use of finite state machines (FSMs) to predict the time to complete tasks given a particular menu hierarchy, and compares offline predictions of task performance with real data in a spelling task. Chapter 5 aims to explore the possibility of abstracting a user's control characteristics from a typical calibration task to predict performance in a novel control paradigm. Online simulation encompasses a range of techniques from low-fidelity prototypes built using paper and cardboard, to computer simulation models that aim to emulate the feel of control of using a BCI without actually needing to put on the BCI cap. Chapter 4 details the development and evaluation of a high fidelity BCI simulator that models the control characteristics of a BCI based on the motor-imagery (MI) paradigm.

The simulation tools and techniques can be used at different stages of the application design process to reduce the level of involvement of end users while at the same time striving to employ UCD principles. It is argued that prioritising the level of involvement of end users at different stages in the design process is an important strategy for design: end user input is paramount particularly at the initial user requirements stage where the goals that are important for the end user of the application can be ascertained. The interface and specific interaction techniques can then be iteratively developed through both real and simulated BCI with people who have no or less severe physical disabilities than the target end user group, and evaluations can be carried out with end users at the final stages of the process. Chapter 6 provides a case study of using the simulation tools and techniques in the development of a music player application. Although the tools discussed in the

thesis specifically concern a 2-class Motor Imagery BCI which uses the electroencephalogram (EEG) to extract brain signals, the simulation principles can be expected to apply to a range of BCI systems.

Contents

List of Tables

List of Figures

Acknowledgements

This thesis would not have been possible without the help of a substantial number of brains. The insight and expertise of my supervisors Prof. Roderick Murray-Smith and Dr. John Williamson have been indispensable in shaping and driving the work forward; thank you for your guidance and patience all the way. To each person in the TOBI team, thank you for a very memorable and unforgettable four years. Special thanks to Michael Tangermann for his advice and help with the simulation work, Robert Leeb for the input and discussions, and Michele Tavella, Luca Tonin, Serafeim Perdikis, Andrew Ramsay and Daniel Boland for providing code, technical support and companionship.

Much of the data collected is also thanks to Serafeim Perdikis, who provided access to his own experimental data in Chapter 3 and collected some for mine in Chapter 4. Mirco Musolesi and Manlio De Domenico pointed me in the direction of the IAAFT algorithm used in Chapter 4, discussions with Harold Thimbleby and Daniel Boland helped to shape the experiments in Chapter 3, and Simon Rogers' patient explanation of various Bayesian-related things has been invaluable. Julie Williamson and Marilyn Mcgee-Lennon provided sound advice for carrying out user studies and experiments. Thank you to all the TOBI partners in WP 10 who provided feedback for the design of the questionnaires and collecting data for the work done in Chapter 6. In particular, it was a privilege to work with Angela Riccio, Martin Rohm, Lorenzo Desideri, Matteo Rimondini and Massimiliano Malavasi. Chris McAdam, Lauren Norrie and Daniel Boland proofread parts of the thesis.

Thanks to all who participated in the experiments.

Many people were part of the journey: thanks to the past and present office mates especially Roseanne, Huda, Mozghan, Daryl, Danny and Lauren for listening and just being there; the GIST and IDI groups for lunches out and drinks and the pub; the TOBI teams at EPFL, AIAS, Graz and Berlin, thank you for your hospitality and care during all the visits; GCCC and the TGIF cell in particular for many weeks and months of encouragement; the GHOP team for your prayers, especially Phillip, Sarah, Susie and Carol; the flatmates Joey and Teresa for many good times and support; Ho Man and Alice, it never ceases to amaze how a brief encounter can go such a long way. I'm sure there are many others I've forgotten to thank especially in the former half of the studies, thank you all. To my extended family, Oz, mom and dad, thanks for everything. And most importantly, to the God who makes all things right and beautiful in His own time, thank You for Your grace and fellowship.

List of contributing publications

M. Quek, J. Höhne, R. Murray-Smith, and M. Tangermann. Designing future BCIs: Beyond the bit rate. In A.Z. Brendan, S. Dunne, R. Leeb, J.R. Millan, and A. Nijholt, editors, *Towards Practical Brain-Computer Interfaces, Biological and Medical Physics, Biomedical Engineering*, pages 173-196. Springer, 2013. (Invited paper)

M. Rohm, L. Tonin, M. Quek, R. Murray-Smith, J. Millan, R. Rupp. Evaluation of Three BCI-controlled AT Devices in a Highly Paralyzed End User. *TOBI Workshop IV, Sion*, 2013.

M. Quek, D. Boland, J. Williamson, R. Murray-Smith, M. Tavella, S. Perdikis, M. Schreuder, and M. Tangermann. Simulating the feel of brain-computer interfaces for design, development and social interaction. In *SIGCHI Conference on Human Factors in Computing Systems, CHI 11*, pages 25-28. ACM, 2011.

D. Boland, M. Quek, M. Tangermann, J. Williamson, R. Murray-Smith, Using Simulated Input into Brain-Computer Interfaces for User-Centred Design, *International Journal of Bioelectromagnetism*, 13(2):8687, 2011.

M. Quek, J. Williamson and R. Murray-Smith. A 2-class Motor Imagery Classifier Output Simulator. *TOBI Workshop I, Graz*, 2010.

List of terms and abbreviations

AAC	augmentative and alternative communication. 17
ALS	amyotrophic lateral sclerosis. 17
AT	assistive technology. 17
BCI	brain-computer interface. 17
CI	confidence interval. 52
EEG	electroencephalogram or electroencephalography. 17
EMG	electromyography. 28
ERD	event-related desynchronisation. 26
ERP	evoked-response potential. 19
ERS	event-related synchronisation. 27
FSM	finite state machine. 52
HCI	human computer interaction. 22
ITR	information transfer rate. 51
LiS	locked-in syndrome. 17
MAE	in Chapters 3 and 4, refers to the mean absolute error; in Chapter 5, the mean average error. 66
ME	in Chapter 5, the mean error. 87
MI	motor-imagery. 19
MI-BCI	motor-imagery based brain-computer interface. 21
PI	prediction interval. 52
SMR	sensorimotor rhythm. 26

TOBI	Tools for Brain Interaction, the EU-FP7 project within which the work for the thesis was carried out. 25
trial	A single attempt to select an interface object or perform a task. 47
TTS	time-to-selection referring to the time taken to make a single decision. 51
UCD	user centred design. 34
WOZ	Wizard of Oz. 39

1 Introduction

Summary. This chapter defines the target user group of brain-computer interfaces (BCIs) addressed in this thesis and discusses the challenges for developing applications under the identified constraints. This leads to the motivations behind the simulation approaches proposed for use the in design and development of BCI applications.

1.1 The need for Brain-Computer Interfaces

A person with a physical disability may be unable to use mainstream technologies such as a mouse or keyboard to operate a personal computer. An input device or software designed to enable such people to use a computer can be called an assistive technology (AT), or alternatively an augmentative and alternative communication (AAC). These include single switches of various forms, head mice, speech recognition systems, eye gaze trackers and even software for touch screen devices.

The most severe form of motor disability is locked-in syndrome (LiS). The term, coined by Plum and Posner in 1996, refers to a state in which a person is almost completely paralyzed yet remains cognitively aware (Khanna et al., 2011). In this condition known as *classical* LiS, vertical eye movements, including blinking, are possible. In *incomplete* LiS, some additional residual muscle movement such as in a finger, toe or head has been recovered, while in *total* LiS, the person has lost control of even eye movements (Bauer et al., 1979). Diagnosis of cognitive awareness can be obtained by manual cognitive assessment and is sometimes evidenced by neuroimaging techniques such as magnetic resonance imaging (MRI) or electroencephalography (EEG). Common causes of LiS are due to a lesion in the pons in the brainstem and neuro-degenerative diseases such as in amyotrophic lateral sclerosis (ALS) (D. Lule, 2009). Detailed discussions of the etiology and complexities of LiS can be found in Patterson and Grabois (1986); D. Lule (2009).

Contrary to intuition, a large proportion of people who find themselves in such a state of being are willing and able to continue living for many years with a good quality of life (Doble et al., 2003). However, empowering people to communicate is essential to maintain a good quality of life (Kübler et al., 2001). For persons with classical LiS, the most common and arguably the most efficient means of communication is via eye blinks or eye gaze to a care giver or other human being. One can either communicate 'yes' or 'no', or spell words by indicating the desired letters on an alphabet board. The invention of the personal computer

and the Internet also improves the quality of life of people who would otherwise be even more isolated from the rest of the world. Eye gaze trackers and eye blink technology allow some people access to computers, and persons with incomplete LiS can use a single switch device (Betke, 2010). However, for people with total LiS, it is impossible to communicate using any overt muscle movement.

A brain-computer interface (BCI) is a system that aims to extract a user's intention whilst bypassing the normal modality of physical movement by measuring and analysing brain signals. This thesis focuses on BCIs whose purpose is to enable people with severe physical disabilities such as LiS to independently use a personal computer or other machine. The system can either be an end user's sole means of digital interaction or communication, or it can be used as an alternative channel to be used during occasions of muscle fatigue Millán et al. (2010). Such a system has the potential for a wide variety of applications for improving the quality of life for people with LiS. The uptake and acceptance of such a system depends on the capabilities, preferences and motivation of an individual (Kübler et al., 2001).

Other definitions and uses of BCIs are not discussed here, although some of the tools described may be useful for such systems. These include gaming applications for healthy people and BCI as implicit interaction where some aspect of person's mental state is monitored in order for the system to adapt to changing user needs (e.g. detection of alertness to compensate for driving in the case of fatigue) (George and Lécuyer (2010), Zander and Kothe (2011)). Training in biofeedback has also been used to help mood regulation for children who have ADHD (Lofthouse et al., 2012).

1.2 Overview of Current BCIs

This section briefly describes the current state-of-the-art of BCI research in general, drawing attention to the BCIs used in the current work. A more complete overview of the different types of BCIs can be found in Dornhege et al. (2007); Wolpaw et al. (2002).

BCIs can be classed into those that use either invasive or non-invasive neuroimaging methods. Invasive methods involve either direct recording of neurons by implanting electrodes into the brain or the use of electrocorticography (ECoG) which detects the summed electrical activity of thousands of cortical neurons (e.g. Hochberg et al. (2006)). Non-invasive methods encompass those neuroimaging methods that do not require surgery. Functional magnetic resonance imaging (fMRI) and near infrared spectroscopy (NIRS) work by measuring the blood oxygen levels in different areas of the brain, and EEG is similar to the ECoG in that it records the synchronous electrical activity of large populations of cortical neurons.

Birbaumer (2006) reported that people with end-stage ALS are more willing to use non-invasive BCI: out of 17 people asked, only one was willing to try invasive BCI methods despite being informed of the possible improvement in speed and accuracy. For this user

group, speed of communication is evidently not an important factor as to be able to communicate at all would be a significant improvement in their quality of life (Birbaumer, 2006).

Of the non-invasive methods, the EEG has the advantages of being one of the most portable, is relatively inexpensive and has the best temporal resolution, and is thus currently a viable system of choice to use for a BCI. Between 2 and 128 electrodes are attached to the scalp via a specialised cap which holds the electrodes. Gel is applied between each electrode and the scalp to reduce the impedance of the signal. Although dry electrode caps are commercially available, within the BCI research community it is generally accepted that the current technology does not provide an adequate signal-to-noise ratio for BCI use, especially for the target end user group.

Regardless of the specific technology used, the BCIs used in this field of research follow the functional model shown in Figure 1.1 and formalised by Mason and Birch (2003), in which relevant features from the raw brain signals are extracted and translated, producing a signal which is used as input to a *control interface*, whose output in turn drives a device or software application. The control interface is defined as a mapping between the signals from the *feature translator*, or translation algorithm (in the BCI used in this thesis, a machine learning classifier), to a control signal that can be understood by the application or device. This may be discrete outputs such as 'top' or 'bottom', or continuous control signals such as the output of a probabilistic classifier.

A *BCI paradigm* defines how and what mental states are detected by a BCI system and used as a person's intention. *Stimulus-driven*, or *synchronous*, BCIs are those which present interface objects as stimuli to the user. In an evoked-response potential (ERP)-based BCI, the user is asked to pay attention to the desired object, and a characteristic brain signal correlated with the timing of the presentation of the stimulus is used to infer the desired object. The most commonly used of these is the P300, which is a positive peak in the EEG that can be detected 200–700ms after the appearance of rare auditory (Klobassa et al., 2009), visual (Nijboer et al., 2008b) or tactile stimuli (Aloisea et al., 2007). A response to flickering visual (SSVEP) or tactile stimuli (SSSEP) whereby the EEG oscillates at the same frequency of the stimulus can also be used (Beverina et al. (2003), Zhang et al. (2007)).

Self-paced, or *asynchronous*, BCIs involve a user voluntarily producing 2 or more mental states which can be separated by a machine learning classifier. The promise of such BCIs is that the timing of the interaction is controlled by the user rather than being dictated by the system. Each mental state is mapped to a particular function of the control interface. The most common of these is the *motor-imagery* (motor-imagery (MI)) paradigm, where the imagination of movement of different body parts are mapped to different controls in the control paradigm. For example, imagination of the right hand might be mapped to selecting the top half of the screen. Although other mental states have been explored (e.g. Friedrich et al. (2012)), most of the research has been carried out with MI as there are well-defined placements for the electrodes which reduces the number required, and mapping these mental

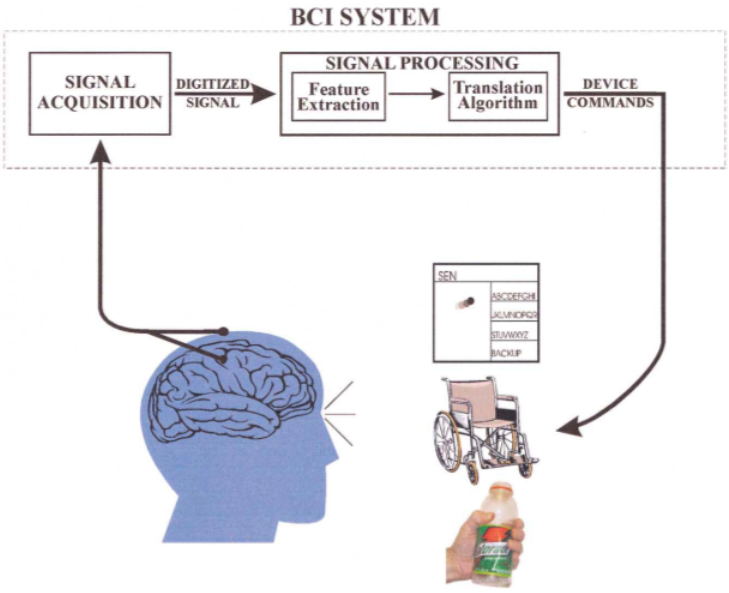


Figure 1.1: Overview of any BCI system. A neurophysiological signal (usually monitoring either electrical activity or blood oxygen level) must first be measured. Features are extracted from this and used to provide evidence of a particular mental state. The system's beliefs are then fed back to the user and may be used to operate a device or control a computer application. From Wolpaw et al. (2002).

states to control objects such as left, right, up or down is seen as being more intuitive than with other mental states such as imagining a cube rotating or doing complicated sums in one's head.

It should be noted that although the term 'self-paced', or 'asynchronous', has been used in the BCI literature to describe such systems, this is somewhat a misnomer as in reality, a truly self-paced BCI where a person can choose not to interact with the system is difficult to achieve. Instead, the timing of the interaction is controlled by the system, which provides cues to signal when the user should perform the mental strategy to control the interface, and breaks to allow the user to rest and plan their next action. This is analogous to scanning based interfaces that are typically used with single switches or single keyboard input. Common to both interaction techniques, the system timings are customised for individual end users.

Until now, stimulus-driven BCIs have been more successful than self-paced BCIs. Nijboer et al. (2010) showed that a higher bit rate was achievable in persons with late-stage ALS for a P300 paradigm than a MI paradigm. In 2009, the first commercial BCI spelling system designed specifically for end users with severe motor disabilities, Intendix,[1] was released. This uses the visual P300 paradigm for an application which allows spelling, sending an email message, controlling the environment, access to a couple of games and a

[1] http://www.intendix.com/

brain painting application. Such a commercially available system is currently not available for motor-imagery based brain-computer interface (MI-BCI).

Although the proposed methodology of design and development of BCI applications described in this work may be applied to stimulus-driven BCIs, the simulation tools have been developed specifically for MI-based BCIs. Self-paced BCIs have the advantage that the user potentially feels more in control of their input to the system, the control input (i.e. the mental states) is independent of the feedback modality, and the feedback is less annoying for users. The complexity of the control in comparison to stimulus-driven BCIs also make it a more challenging area of research. Details of the control characteristics pertaining to MI-BCIs are discussed in later chapters.

1.3 Difficulties in Developing Applications for BCIs

In 2008, Lécuyer et al. (2008) commented that there had been little uptake of HCI research into BCI applications. While the BCI research community continues to develop and improve feature extraction and classification methods, very few BCI applications actually exist. The ones that do exist often feature very basic features or clunky interfaces which may be difficult for the user to learn.

Several reasons contribute to making it difficult to design and develop applications for BCIs. Firstly, access to end users is very limited as there are few of them and it may take weeks to organise interviews or user trials with them. It may take a long time for them to articulate their desires and wishes, and they can tire quickly. Yet, being able to involve end users in the design process is especially important as the gap in knowledge of what users want is greater for designers designing for people who have special needs than in designing for people with no major disabilities (Thimbleby, 2008).

Secondly, heterogeneity among the end user group is large as people have different capabilities and different needs and desires. In terms of sensory capabilities, an individual may have combinations of visual, auditory or tactile sensitivity which any communication solution must be adapted for. For example, a person who is not able to control their eye gaze may not be suited to using a visual ERP-based BCI, but may still be able to have enough control to view a display on a large screen. In terms of user requirements, people will inherently have different priorities. For example, one person who was allowed to communicate using a BCI for the first time asked, 'Why am I wearing such an ugly shirt?', which surprised her therapists (Kübler et al., 2001). These factors make it difficult or even impossible to design a one-size-fits-all application or device for end users.

Large individual differences also exist in the control characteristics of BCI. For example, the phenomenon of 'BCI illiteracy' (Blankertz and Vidaurre, 2009), where a user is unable to produce mental states that can be reliably detected by the BCI, is found for about 20% of the general population for MI-BCIs. Thus, customisation of the application and control

parameters for each individual person is required, and there is no guarantee that a user will be able to use a BCI despite many sessions of training.

Individual differences contribute to the time-consuming nature of carrying out usability tests and experiments with people, and even more so with end users. At least for MI-BCIs, training over a few sessions, ranging from one or two hours each, is required. For each session, some time has to be allocated to fit the BCI cap, apply gel and clean up afterwards, and because the communication rate is slow, the actual user study can take a few sessions of a few hours each. The time-consuming nature of BCI research should be highlighted as it is common in human computer interaction (HCI) research for experiments to be quickly carried out. In a 2007 workshop entitled 'BCI meets HCI',[2] HCI researchers learned that BCI research is a slow process. Many of the ideas proposed and research questions raised by the HCI researchers were received with interest; however it was pointed out that each new idea would take months to plan and run an experiment for.

Finally, the entry level into development of BCI applications is high. The main reasons for this are the high cost of purchasing the necessary hardware and the knowledge or skill base required to process and use the brain signals. This is exacerbated by the fact that BCI control signals have not been standardised: the application cannot easily be separated from the specific control signals expected by the system. Thus, it has generally been difficult to develop the BCI application separately from the control signal. As the control characteristics of BCI input are qualitatively different from other input methodologies, guidelines (e.g. Gajos (2010); Doherty et al. (2001)) are helpful but not sufficient for building BCI applications. Typically, the BCI application is developed by the same engineering team which develops the algorithms for feature extraction and classification. Without the resources to apply HCI and design principles to applications, it is difficult to develop applications that enhance the user experience. Experience with other BCI researchers working in the field also suggests that time can be wasted when a system failure occurs in a full BCI trial that could have been uncovered prior to placing a real brain in the loop.

This thesis aims to show that simulation techniques can be used to aid design and development of BCI applications, lessening the impact of the problems described above. McFarland et al. (2003) postulated that simulation could be useful for designing and developing BCI applications rather than inferring anything about control of new paradigms; however this has not really been investigated in the BCI literature.

1.4 The Role of Simulation and Modelling in Design and Development

A model of a system, artifact or environment is a simplified representation that captures its essential characteristics for a specific purpose. A simulation is the operation of the model,

[2] IDIAP Research Institute, Martigny, Switzerland, 2007

where the intention is to draw conclusions, qualitative or quantitative, about the behaviour or properties of the system. Simulation is used in a wide range of contexts where it is either more expensive, impossible or impractical to investigate or test a real system (Maria, 1997). It is also useful in situations where it is more expensive, difficult or impossible to build a mathematical model of the system that can be analytically solved.

This thesis distinguishes *offline* and *online* simulation. In *offline* simulation, a computer model of the system is run to explore or predict the behaviour of a system. Some applications of this include investigating natural phenomenon in biological and physical sciences, and developing systems in business operations and networking. In *online* simulation, a human being may be placed in the loop to interact with the system. An example application is a flight simulator which is used to train pilots before actually flying an airplane. Online simulation can also be defined as a real-time visualisation of a system being run automatically, such as in a visualisation of weather prediction.

In mainstream HCI, models are widely advocated as being useful for design and development. Kieras (2003) explains that model-based evaluation is needed to provide insights into usability before testing with end users. Besides saving time and development costs, models of human performance can contribute to scientific knowledge of how people interact with computers. He uses the analogy that software engineering needs to become more like engineering a bridge: the developer should know that a system is going to be stable before actually building it. In this respect, simulation techniques can be used to predict and optimise the performance of a system, and to check for safety and correctness.

However, model-based evaluation is probably not as crucial in mainstream consumer applications for the average healthy user as it is for people with severe cognitive or motor disabilities. This is because easier access to a healthy target user group allows for rapid prototyping in iterative design. In contrast, it can be more difficult to access people with disabilities and a user study can take longer and is more effortful for the participant. Inclusive design aims to make it possible for mainstream applications to be used by people of all abilities. To this end, good models and guidelines must be made available to designers and developers. Offline simulations can be used to automatically check the properties of a system, but their usefulness is necessary limited as there is no validation by an actual user. If used correctly, online simulations can be used to enable designers to experience a system from another point of view. By partially experiencing someone else's situation, they may be able understand the issues better and thus come up with suitable solutions (Poulson et al., 1996).

Because of the extreme nature of BCI control characteristics and user groups, simulation techniques have the potential to be extremely useful in reducing the time spent on virtually every stage of the design and development process. In requirements gathering, prototypes can be used to elicit end user reactions to scenarios. Simulation can help to reduce the time costs of application development by making predictions about performance and enabling user testing without the costs of setting up the BCI, and lower the threshold of entry into

designing applications. Although simulation cannot completely replace end user involvement, the aim is to show that the benefits of simulation outweigh the costs of developing the tools in the first place. Chapter 2 proposes a design process through which simulation techniques can aid in the design and development of BCI applications.

1.5 Overview of Thesis and Contributions

Designing applications for severely disabled users using a Brain-Computer Interface (BCI) as an input mechanism is difficult due generally to the time consuming nature of end user trials and the heterogeneity of the target population. This thesis presents two simulation techniques that have been explored in order to inform the design and development of BCIs: namely *offline* simulation which aims to predict the performance of a BCI, and *online* simulation whereby the subjective feel and control of the BCI is captured in order to allow exploration of new paradigms and applications. The main contribution is the introduction of a design process through which simulators are used at different stages to design applications for people with Locked-in Syndrome (LiS).

Chapter 2: Describes the uses of simulation for the purposes of design and development in HCI, AT and BCI research, placing the current work in the context of the state-of-the art techniques. Control characteristics of BCIs are described, followed by a proposed design process for a BCI application which uses simulation at different stages to inform the design or understand underlying principles of interaction.

Chapter 3: By representing menu hierarchies using finite-state machines, graphical modelling tools can be developed that are easy for designers to work with. Simulations of item selection for menus in which there is a uniform probability of selection show that it is important to match the performance and control characteristics of users with the error correction mechanisms. Although this information in itself is not new, as others have demonstrated this using various techniques, the specific findings with regard to the value of the specific error correction methods menu hierarchy have not previously been published. In addition, there have been no comparisons between predictions based on the calibration trials and actual online performance. To evaluate the models, we thus compare offline predictions of task performance based on the calibration tasks, with control in an actual application setting. The contributions made here are the modelling of menu hierarchies in terms of time taken to achieve tasks and the comparison between offline predictions of task performance and actual online performance.

Chapter 4: This research also provides the first attempt to explicitly model the classifier output of a BCI system for the purpose of design and development of applications. A model of the time series is useful as it allows the feel of the control characteristics to be captured in the online simulation mode. Additionally, it can replace the BCI signals as real input into the BCI application being tested. Several methods were used to model and generate the output of the system. Evaluation of the models using quantitative measures indicates

that different techniques are useful for modelling different characteristics of the system. Evaluation with BCI experts indicated that the model was sufficient for use for BCI input. Healthy users also provided qualitative feedback on comparing their use of real BCI with simulated BCI. The simulator can thus be used to explore applications and paradigms in an 'online simulation' manner.

Chapter 5: Simulation models were applied to the rotate-extend (REx) selection mechanism. This is a novel paradigm for 2-class motor imagery that is a generalisation of the original Hex-O-Spell interface. Here, one mental class is used to rotate an arrow round the centre of a wheel, while the other class is used to extend the arrow in order to select a segment of the wheel. Offline simulation was used to estimate the expected performance of the REx paradigm from binary trials for a given user. The results showed that combining the results from several models can provide predictions that better match the real data. The approach, whereby calibration data from individual users are used to predict performance in a novel paradigm provides motivation for further research into improvement of simulation models for predicting individual user performance in novel paradigms.

Chapter 6: A case study of designing and developing a music player application is presented. This applies the proposed design process in Chapter 2, where the fidelity of developed prototypes in each design stage is traded off with the number of end users close to the target group of end users. Findings across stages and across users are drawn together to provide insights into a music player that might be developed for an actual end user with LiS. The benefits and limitations of the design process are discussed.

It is worth noting that the work in this thesis was completed within an EU-FP7 project, Tools for Brain Interfaces (TOBI).[3] The choice of using two classes for the motor-imagery paradigm was a commitment made early on in the project, while the BCI system used in the experiments was provided by the CNBI group in EPFL, a partner in the project (details in Section 3.2). This allowed the design and development of applications without requiring to develop the system from scratch. As such, the signal processing, feature extraction and classification methods used were determined by the system, which generated values from a probabilistic classifier at 16Hz. These values were output over the TCP/IP network according to a project-defined protocol and used by the author in the development of experiments, selection mechanisms and applications. The choice to use the same BCI system, developing applications separately from the low-level processing, meant that other partners and end users in the project could simply use the applications once a user had been trained to use that system. Data from experiments involving people with disabilities, namely the validation of offline simulations in Section 3.6 and the user evaluations in Chapter 6, were either obtained from, or carried out in collaboration with, BCI researchers and AT professionals; more details are explained in the relevant sections. The experiments involving only healthy participants in Chapters 4–6 were designed and carried out by the author.

[3] www.tobi-project.org

2 Background

Summary. This chapter begins with a brief background on motor-imagery based brain computer interfaces (MI-BCIs), as this is the main focus of the simulation tools described in the next chapters. It then describes the control characteristics of state-of-the-art BCIs, emphasising those based on detection of voluntary ('asynchronous') mental state detection. A literature review of simulation techniques used in HCI, BCI and assistive technology (AT) research draws attention to the similarities and differences between BCI control characteristics and that of other input modalities, and positions the current simulator tools in the design and development process of a BCI application.

2.1 Motor-Imagery Brain Computer Interfaces

MI-BCIs are a subset of the class of BCIs that are based on voluntary potentials; that is, the system input depends on its ability to detect mental states that are voluntarily produced by the user, rather than an EEG artifact produced in synchrony with an external stimulus. The EEG measures the voltage potential difference between each electrode in the EEG montage and a reference electrode that is usually attached to an electrophysiologically neutral part of the body, such as the ear lobe. A standardised system called the International 10-20 system allows for placing the electrodes strategically depending on the mental strategy or strategies that have been chosen to operate the BCI. For the system used in the course of this research, 16 electrodes are used; however, as few as 3 (Scherer et al., 2007) for a subject-specific choice of location and as many as 128 (Blankertz et al., 2006a) have also been reported in the literature.

Large numbers of neurons firing at the same time produces oscillations in the EEG signal at a particular location of the brain. These oscillations are characterised by the amplitude in different frequency bands. Sensorimotor rhythms (SMRs) are oscillations in the EEG located in the sensorimotor areas of the brain, which can be represented by the homunculus–the 'little man in the brain' (Bear et al. (2006), Figure 2.1). These *motor neurons* are associated with the motor control or sensation of different body parts. During rest, when no motor activity is present, large numbers of neurons are firing in synchrony give rise to oscillations in the 7-13Hz range (α-rhythm) and the 15-30Hz range (β-rhythm) (Pfurtscheller and Neuper, 2010). Actual or imagined movement of specific body parts disrupts the synchrony, which is referred to as event-related desynchronisation (ERD), a reduction of the amplitude of the related frequency bands. An increase in the synchrony is referred to as event-related

synchronisation (ERS). As the ERD and ERS can be localised according to certain body parts that are being imagined, this forms the basis of classifying the mental states. As SMR activity has been well established in the literature, MI-BCI has been the most popular choice for mental-state based BCIs.

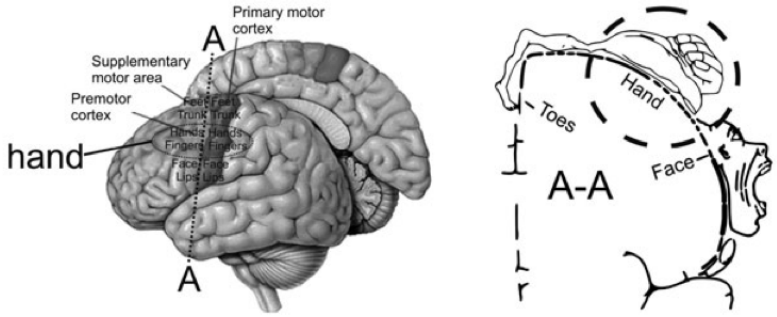


Figure 2.1: Left: sensorimotor areas of the brain. Right: the motor homunculus, or 'little man in the brain', is a representation of the area of the primary motor cortex dedicated to different parts of the body. From Pfurtscheller and Neuper (2010).

Still, around 20% of people are currently known to be 'BCI illiterate' with MI-BCI (Guger et al. (2003), Fazli et al. (2009)), where they are unable to reach an acceptable level of performance for communication. Although the reasons for this are still under active investigation, recently some inroads have been made into discovering why this may be the case. One reason for an inability to control a MI-BCI occurs when the correct mental strategy is not used. The type of feedback presented as well as inter- and intra-subject variation may also affect the ability to operate an MI-BCI; Grosse-Wentrup and Schölkopf (2013) presents an up-to-date review of the current literature on performance variation in BCI.

2.2 Control Characteristics of a BCI

BCI control characteristics are different from other input modalities. In this section, similarities and differences between different modalities are discussed. The aim is to provide a background for understanding the later literature review of tools and techniques that have been used in mainstream HCI, AT and BCI research, and in the development of the simulation tools used in later chapters.

2.2.1 Required Motor Capabilities for Operation

Apart from BCI, most available input techniques for communication require motor input. Mainstream technologies for healthy users generally require the use of the hands for input, and other modalities are rarely used. An exception is speech recognition software, which has been used increasingly by the general population as the technology improves. There is a wide range of available ATs to enable a person to communicate. If a person has residual use

of any muscle, a single switch device can be can be operated. Examples are a single shoulder muscle or head movement, a breath or mouth-operated switch, or electromyography (EMG), which can be used to detect even weak muscle activity. Other examples of ATs for physically disabled users include arrays of switches, mouse emulators which can be controlled using a mouth stick, and eye gaze technology which tracks eye movements in order to allow a person to select objects on the screen. The ATs are usually selected and (sometimes ingenuously) customized to a person's preferences and individual capabilities.

For ERP-BCIs based on visual attention, good control of ocular muscles should be present. This is comparable to eye gaze technology, which requires good control of ocular muscles for fixating on a particular point on the screen in order to select a particular interface object. However, the BCI research community has been developing ERPs using auditory evoked potentials (Höhne et al., 2011) as well as covert attention (Aloise, 2012), which do not require the ability to fixate on a particular object; thus no motor capabilities need to be present. A visual ERP system may be more usable for people for whom eye gaze technology does not work, due to the inability to control one's eye gaze sufficiently (Nijboer et al., 2008b). For oscillatory-based BCIs, motor control is not necessary. A hybrid BCI has been defined as a human-computer system that uses at least one intentional BCI signal as input (Pfurtscheller et al., 2010). For example, the system can use different physiological signals or motor input either to augment or substitute the BCI signal. Thus, a person could use a single switch or other modality to access a computer, switching to BCI mode on the onset of fatigue. The time of switching could either be manually controlled or automatically determined by the system based on its assessment of poor signal quality.

Thus, the potential benefit of BCI is that people who are completely paralyzed, and additionally do not have good eye gaze control, are nevertheless able to communicate simply by using thought alone. One limitation of this is that there is no proprioception in BCI control; that is, the user is unable to get feedback from muscles and joints about exactly what they are doing. Thus, feedback from the system is necessary to fulfil this role.

2.2.2 Required Sensory Capabilities for Operation

BCI systems share a similar trend with most mainstream consumer technologies in that most interfaces have been developed to display their output visually onto a digital screen. Yet systems have also been built for auditory output and tactile output. Most commonly, screen readers such as JAWS[1] for Windows and the in-built screen reader for the Macintosh OS are used effectively by people who are blind. People who are locked-in may lose ocular control to the point that they are unable to see, and thus it is important for BCI research to develop in terms of auditory and tactile output. A review of the small number of current studies that develop BCI systems that do not rely on the visual modality can be found in Riccio et al. (2012). As the output of the system is generally independent of the input modality (apart from with ERP-BCIs where the link is inherent), the same controls need

[1] http://www.freedomscientific.com/fs_products/software_jawsinfo.asp

only to be re-mapped to the relevant application controls, tailoring the system's control and output to the particular modality.

2.2.3 Degrees of Freedom

The degrees of freedom allowed for a single selection varies for different input technologies. Assuming that the size of a typical on screen button is $100 \times 25px$, a single click of the mouse is able to select one out of 300 items for a screen with a resolution of $1024 \times 768px$. For visual ERP-BCIs, typically 2-36 stimuli are reported in the literature, while for eye gaze technology a full size keyboard (approx. 70 keys) may be placed on the screen. In both cases, the number of objects that can be placed on the screen for selection depends on the degree of ocular muscle control the user has.

In terms of the number of mental states used in oscillatory-based BCIs, between 2 and 6 classes have been reported in the literature (e.g. Bashashati et al. (2007), Obermaier (2001), Doud (2011)). The choice of number of classes is dependent on the choice of system and the user's preference and ability. 2 classes are most widely reported in the literature, allowing for binary (e.g. 'yes', 'no' or 'left', 'right') responses. Obermaier (2001), Dornhege (2006) and Kronegg et al. (2007) demonstrated that for the levels of accuracy that are currently achievable, 2–4 mental classes are optimal depending on the user and the BCI system used. However, increasing the number of classes may not improve the ITR significantly enough to warrant the extra efforts involved in developing an application interface to account for the higher degrees of freedom. The number of degrees of freedom can also be increased by increasing the control paradigm. McFarland et al. (2003) conducted an experiment where users controlled the vertical position of a cursor moving horizontally from left to right, with the screen divided into 2, 3, 4 or 5 target segments. For most users, the optimal ITR was achieved when 4 targets were used. Besides increasing the number of mental states used to control the number of available degrees of freedom, 2-D (Wolpaw and McFarland, 2004) and 3-D (McFarland et al., 2010) control has also been achieved by training the subject to regulate specific oscillatory features in a positive (ERS) or negative (ERD) direction, increasing the number of dimensions as the user gains control over the lower dimensions.

Although some systems operate by distinguishing between two or more mental states that are intentionally produced by the user, attempts have also been made to detect single mental states from rest (Pfurtscheller and Solis-Escalante, 2009; Fazli et al., 2009), allowing for a 'brain switch'. This is analogous to using a single switch system, assuming that the user has some control over the timing of the system. In both cases, a hierarchical interface is used to divide the selection space. For example, it may take several selections to select a single letter of the alphabet.

2.2.4 Selection accuracy and time-to-selection

Selection accuracy is the percentage of time the user is able to select what they intended to select, while the time-to-selection refers to the time it takes to make a single selection. Together with the number of degrees of freedom, these limit the communication rate achievable through the system, and are important to bear in mind in designing applications for BCI. For a 2-class BCI, a selection accuracy of at least 70% is commonly taken to be the minimum for reliable communication to occur (Blankertz et al., 2008; Kübler et al., 2004), while Müller-Putz et al. (2008) showed that for an experiment of 20 trials per class in a 2-class system, the upper 95% confidence limit of chance selections is 70%.

Selection accuracy for healthy users using mainstream input technologies (mouse and keyboard) is assumed to be reasonably high, with errors being due to user errors such as slips and mistakes (Norman, 1998). Slips occur when the user intends to make the correct action for the desired goal but inadvertently produces the wrong motor action, while mistakes are due to the user actively selecting the wrong action, perhaps due to a misunderstanding of the consequences of the inputs. In this sense, BCI shares a common ground with input technologies based on machine learning, such as speech and gesture recognition, where instead the errors are due to a mismatch between the user's intention and the system's belief. With such systems, the inability of the system to detect the user's intention can be due either to noise in the environment (e.g. electrical noise surrounding a BCI), or the inherent variability of human performance (the user does not do exactly the same thing each time). With BCIs, noise also occurs due to the presence of neural activity that is not related to the brain signals of interest. Frustration occurs when the system does not detect the user's intention, and there is frequently a mismatch between the user and system's (mental) models of the features that are used to distinguish the relevant instructions to the machine. Thus, the user often cannot figure out why the system does not recognise their input, and this leads to further stress and frustration.

The time taken to make a single, discrete, selection affects the total amount of time required to complete a task. In HCI, well known models of movement time (Fitts' Law (Fitts, 1954)) and time taken to make a decision (the Hick-Hyman Law of choice reaction time, (Hick (1952), Hyman (1953))) can be used to predict the time it would take a user to makes selections and thus complete intended tasks. In BCIs, the time taken to make a selection depends on the choices made by the system designer with regard to the length of time allocated for the system to gather evidence about the user's intention, and whether this is fixed or variable. In cue-based or synchronous interfaces, an addition is the time between selections which are for the user to have a break or decide on the next selection. For ERP-BCIs, the time taken to make a selection has usually been fixed, based on the chosen length of time or number of stimulations presented to the user. Recently, dynamic stopping techniques (Schreuder et al., 2009) have been employed where the stimulation is stopped when the system deems that enough evidence about the user's intention has been gathered. For MI-BCIs, both fixed time and variable time selection methods are employed. Typically

for the former case, the participant is asked to control a feedback cursor to a target position. At the end of the allocated time for evidence accumulation, the selection is taken to be the one closest to the position of the cursor. In the latter case a decision is made when the accumulated output reaches a threshold (i.e. the feedback cursor reaches the target).

For MI-BCIs this may be even more extreme and frustrating, as often the user has no way of knowing if the uncertainty in the system is due to their own inconsistency in imagining a movement that can be easily distinguished by the system. As the evidence is accumulated over time, it may be that repetitive movements (e.g. repeatedly opening and closing the hand) are easier to detect rather than a single movement (e.g. prolonged clenching of the fist), but the noise in the system does not allow the user to easily figure it out.

2.2.5 Asynchronous control

Asynchronous control refers to how much freedom the user has to decide when to interact with the system. A continuum exists in BCI between fully synchronous and fully asynchronous interaction. In fully synchronous interaction, the time of interaction is fully determined by the machine. No other input technology shares the same level of synchronicity as ERP-BCIs, as the user's neurophysiological changes are an unconscious response to the system generated stimuli. In fully asynchronous interaction, the user is able to determine exactly when he wants to interact with the system. This is the case of most common input devices such as keyboard or mouse input. Other assistive technologies may lie somewhere in between the extremes, for example with a single switch scanner, the system scans through in its own time and the user has to make a click at the right time.

Achieving a truly asynchronous control with BCI is an active area of research (Fazli et al. (2009), Millán et al. (2008)). This aims to identify 'idle' or 'rest' states in which the user can voluntarily choose not to interact with the system. The BCI used in this research can be thought of as a semi-synchronous protocol, where the system dictates when the user can provide commands to the system, but there is a variable timing as to when the user makes a selection (see Chapter 3.2 for details).

2.2.6 Input-Output Delays

Delays are inherent in any man-machine system and may arise from the either the system or the user. Williamson (2006) (pp. 31) provides a succinct overview of the sources of delays in a typical man-machine system. These influence the delay the user feels between intending or performing an action, and the effect of that action being presented by the output of the system. Delays can create uncertainty and frustration for a user, and affect user performance in manual control systems.

Systems which incorporate information over a period of time, and optionally require high computational power for processing, such as gesture or speech recognition, have inherent

delays which are noticeable to the user. Transmission delays can occur in systems where signals have to be transmitted over long distances, such as in teleoperations or telerobotics (Smith and Christensen, 2009). These may create similar effects as in self-paced BCIs, as the user has to wait for a period of time before the system responds or is even able to acknowledge an input. People can learn to compensate for delays to an extent; however if the delay is too long or too inconsistent, this is no longer the case. With the BCI used in this research, the delay is likely to be around 0.5s at minimum, which occurs due to the 1s time window within which EEG signals are collected (Millán et al., 2008).

Delays can also arise from the user-end of the interaction. For people with physical disabilities, it can take a longer time for the user to be able to convey their desired input to the system. For example, tremors may cause a person to take a longer time to press a button. Yet this differs from BCI as there is usually some proprioceptive or visual feedback, and it is less ambiguous when they have provided input to the system. The system attempts to substitute for this lack of proprioception by providing feedback in the form of the cursor movement, but because of the delay added to the system and the noise and uncertainty, it can be difficult for the user to use the feedback unless they have fairly good performance (see later chapters for a more in depth discussion on the issue of feedback). It is possible that the lack of proprioception also makes the delays seem longer. Delays specific to asynchronous BCI include the time it takes to switch between mental states.

2.2.7 Noise and Uncertainty

In a motor-controlled system, one is able to see and feel movement one's one movement. While tremors and spasms can occur, such as in Parkinson's disease or cerebral palsy, making it difficult to make finely controlled movements, visual and proprioceptive feedback mean that the 'noise' can be seen and is unambiguous. For other HCI techniques that use artificial intelligence to infer user intention, such as speech recognition or gesture interaction, noise and uncertainty also arise because the system's classification of the user's input may be wrong due to noise or variability. However, here again the user is aware of their inputs and how the system is misreading their intent. Conversely, in MI-BCI, the effect of noise is enhanced in part because of the lack of proprioception: a user who is learning to use a BCI has no idea whether they are reliably producing the correct mental states or signals that can provide useful information to the system for classification.

2.2.8 Stimuli and Feedback

In an ERP-BCI, stimuli is presented to the user at a much higher frequency than needed for other input modalities. As the brain response is coupled to the target stimuli, reducing the time interval between successive stimuli is desirable. Nevertheless, this can be uncomfortable for the user. This issue is related to the synchronicity of the user interface. In a non-ERP BCI, feedback is usually presented to the user in the form of a cursor moving on the screen. The goal is for the user to be in voluntary control of the feedback. The role of feedback in

MI-BCI is an ongoing topic of research (Neuper and Pfurtscheller, 2010).

Stimulus-response compatibility is a concept formalised by Fitts and colleagues from the 50́s (Fitts, 1954), which refers to a consistent and natural mapping of system controls to the user's expectations. For example, a natural mapping for switches controlling the left and right lights in a room is for the left switch to control the light on the left. Incompatibility between what the user and the system expect can be annoying for users in the best case, and have fatal consequences in situations where there is a great degree of risk. The phenomenon relates to self-paced BCI, as the user's mental states (e.g. imagining left/right hand movements) must be mapped somehow to the system controls (e.g. left/right, up/down). It is appreciated that natural mappings should be made as far as possible, and this is a reason for choosing motor imagery as a strategy for BCI control; yet this is not possible even for binary selection using an MI-BCI, as the best strategy for some users is not left and right motor imagination (corresponding to moving a cursor left and right), but left/right hand and foot motor imagination. The effect of this on user learning and ease of controlling a BCI has not been formally investigated. It is possible that slightly unnatural mappings in a binary selection would not affect the user too much as people can learn the simple mapping, but for a larger number of classes where the mental states do not naturally match a system control (such as mental rotation to move a cursor to the left), or for other control paradigms, an overhead at least for learning the mental state-system mapping should be taken into account.

2.2.9 Summary of Control Characteristics

To summarise, in addition to the lower rate of communication than even most ATs, MI-BCIs have additional overheads including the noise and uncertainty due to a lack of proprioception. The various different types of BCI having different properties can be confusing to a new user of ATs, and the control characteristics of the interaction can be difficult to comprehend to a naive user who has not actually used a BCI. Even while watching someone use the system, it can be difficult to comprehend the skill required to operate a MI-BCI. Thus, we argue that simulating the control characteristics of the system can at least inform someone of the nature of using the system. Even as BCI technology continues to improve, simulating the system can help designers, developers and stakeholders to work with and communicate the new technologies.

2.3 Simulation in the Software Development Life Cycle

To simulate is to 'imitate the appearance or character of' an object (Oxford Online Dictionary[2]). To this end, various prototyping and modelling techniques can be placed under the umbrella heading 'simulation', which are used at various stages of the software development cycle, from design to development. This section provides an overview of the ways in which

[2]http://oxforddictionaries.com

different simulation techniques are used to design and develop software applications in HCI, AT and BCI research. The benefits of using these tools, as well as gaps within the BCI literature, are identified. *Offline* simulation is distinguished from *online* simulation, where offline simulation refers to a model that is run at 'computer speed' such that the output are summary statistics of the metrics the user wishes to analyse. In contrast, in an online simulation, the system is run in such a way that that the user can see the changes at each step of the simulation in real time. This can either be passive such that the user only views the simulation once it has been set up, or it can be interactive such that the user is actively involved in engaging with the simulation in real time.

Several themes run throughout the discussion of simulation tools and techniques. The *fidelity* of a simulation is the degree to which it replicates the exactness of the object it models. As a simulation is a model, or simplification, of a real system, it is naturally never an exact representation of a real system. The choice of fidelity at which to build the model is related to the *cost* (at least in terms of time invested) and to the *purpose* of the simulation. There is a trade-off between the cost and fidelity of any model, such that the value of improving the fidelity of the model is justified by the cost of development, up to a point where the return on investment becomes too small to warrant the improvement. It is up to the model user to make a judgement call based on whether a model is 'good enough' for its intended purpose, as a low-cost prototype or simulation is not necessarily less useful than a high-fidelity one depending on the purpose and the stage of development. Identifying the purpose of the system also allows one to isolate the properties of the simulation that need to be focused on. Thus, the fidelity of the simulation may not refer to the whole system, but to specific aspects of the simulation, such as the fidelity of the experience (Buxton, 2007), or the physical design of the system.

Another theme that runs through the discussion on simulation tools and techniques is the tightness of coupling between the user and the simulation. User centred design (UCD) is an approach to design which aims to produce solutions that match the real requirements of end users (see section 2.5). This approach is firmly embedded in the culture of design and development, where the system is designed to create the best possible experience for the user. The simulation techniques can be positioned in a space that represents the tightness of coupling between the user and the fidelity of the simulation (Figure 2.2), which is a useful representation as it shows how a mixture of different types of simulation can be used to engage the user and provide answers to design questions.

2.3.1 Offline Simulation and Modelling

Offline modelling and simulation of human-computer interfaces has been a wide-ranging field of research almost since computers have existed. Formal methods involves representing a system in such a way that its properties can be analysed without having to manipulate the real system (Dix, 2003). The abstraction can take the form of mathematical or computational models whose output can be computed or simulated, or diagrams which can be

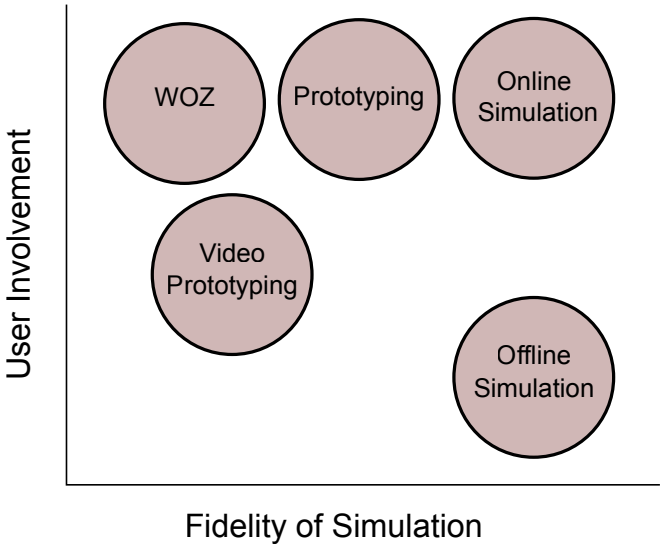


Figure 2.2: Coupling between the user and the fidelity of simulation. In this case, fidelity can either refer to the look and feel of the simulation, or the accuracy of behaviour the system is supposed to model.

automatically analysed or manually stepped through. Applications of offline simulation of human-computer interfaces include checking and accounting for errors and predicting task performance.

Error checking. At the system level, a model of the system can be used to check the correctness of a software design or an actual system. Diagrams such as State-Transition Networks (STNs), Petri-Nets or Harel State Charts, and textual models such as process algebras, are used to model the user interface, which can be used to check that desired properties of the system hold (Dix et al., 2004). For example, in checking for *reachability*, the designer wishes to check that the system allows the user to get from every state of the system to every desired state (Dix et al., 2004). It is also important to check that user inputs made at every state of the application lead to desirable outcomes. Perhaps because applications designed to use a BCI as input are simple due to the limited number of degrees of freedom that are available, such techniques have not typically been employed within a BCI application development context. Text entry programs are also typically built with formal structures, sometimes incorporating language models to increase the communication rate (e.g. Perelmouter (2000)), and thus it may be more difficult or time-consuming to simulate these tasks. Nevertheless, model checking can be used for applications such as those for controlling environment variables like lighting or the television, which involve hierarchical menu structures. As errors are common and can either be due to slips, mistakes, or system misinterpretation as previously discussed, it is important for designers to check that errors are undoable and do not lead to disastrous consequences.

Predicting task performance. Simulation and modelling can be used to predict task performance. Formal modelling in HCI involves developing a mathematical model of the user's interaction with the system, incorporating aspects of the system in order to predict performance measures such as time taken to achieve tasks and how often tasks can be accomplished. The model can be used to provide predictive or comparative measures of performance either through calculation or simulation. Simulation rather than calculation is used where it is either more difficult, costly, or simply impossible to obtain a mathematical model. The cost of development of such models is related to the precision or complexity that one wishes to capture, and is traded off with the level of increasing return.

The Model Human Processor (MHP), first described by Card (1986), is a useful framework for analysing the system from the user's point of view, dividing the user part of the system into the cognitive, motor-behavioural and perceptual aspects. Psychophysical models such as Fitts' law and the Hick-Hyman law of choice reaction time both provide predictive, quantitative measures as well as guidelines for design, while the GOMS models (John and Kieras, 1996) predict task times based on separating tasks into micro tasks and summing the expected time to complete the micro tasks. The Cogtools application (John, 2011) allows one to set up usability tests which predict how long someone will take to complete a task given a GUI layout. This incorporates models that have been developed by psychologists and human factors researchers over many years, rather than requiring the user to have in depth knowledge and implement these themselves. Limitations of these models are that they model expert users and do not take into account user error or learning. The emphasis on the motor aspects of these models does not allow them to be easily used in the context of BCI, where there is no motor control and thus entirely new models of the user input need to be built.

Controversy exists within the HCI community of the value of modelling and simulation for mainstream input techniques, particularly when it comes to modelling of cognitive architectures. It might be argued that usability testing is quicker and more valuable than offline simulation. This may be warranted to some extent with interfaces using input technologies that are quick to implement and interact with in real time, and with users who are easily accessible. However, for developing interfaces for people with disabilities, the lack of immediate access to users, the diversity of issues and disabilities, and the slow rate of communication make offline simulation a valuable tool for designers and developers to build new techniques for interaction and to predict usability for users with particular characteristics. The work by Biswas (Biswas and Robinson, 2008b; Biswas and Langdon, 2011) aims to develop models and simulators for disabled users that can be used to predict task performance in a system by considering the user aspects of the system. Models of motor control of disabled users have been used to optimize applications for single switch scanning. For example, Bhattacharya et al. (2008) identified the nature of errors that motor-disabled users make in single switch scanning tasks, which were subsequently used to predict the performance of different text entry designs. However, BCI-specific models need to be developed as the nature of the interaction and the inputs are different to other input technologies. While

the perceptual models from previous work may be useful, cognitive aspects such as fatigue, stress or motivation need to be investigated further to understand and predict their effect on performance.

2.3.2 Online Simulation and Prototyping

The need to involve the end user in the design and development process, as early and often as possible, is a principle that is firmly embedded in the principles of UCD. Many of the tools and techniques that can be presented to the end user for discussion, exploration and testing are simulations of a future system and can be placed under the umbrella terms prototyping and simulation.

Houde and Hill (1997) defines a prototype as 'as any representation of a design idea, regardless of medium' which has the purpose of answering specific design questions or exploring design options. Thus, a prototype can be represented as a point within a triangular functional space, where its purpose is to investigate some combination of the *role* a system will play in users' lives, the *implementation* of the system (how the functionality should work), and what it should *look and feel* like (Figure 2.3). Identifying the purpose of the prototype helps to narrow down the questions that the designer should ask, and to identify the tools and techniques that should be used. The authors also emphasize that both high- and low-fidelity prototypes can be used at different stages in the design process for different purposes. The prototypes can be used in various contexts such as focus groups, interviews, online studies, surveys, usability studies and field studies.

This section explores different purposes for prototyping as well as tools that can be used for each purpose, discussing their applicability to BCI design. In each of the techniques presented, there can be low-fidelity versions or high-fidelity versions. Lim et al. (2008) suggests that the best prototype for a given situation should be one that is the simplest to implement for its given purpose.

Exploring the role of a system. The role of a system is the part it will play in the life of the user, how the user intends to use it and in what contexts. For example, does a person want to play games with other people, or just by themselves? This can be thought of as the part of 'getting the design right' (Buxton, 2007) in the design process. Designers can use *scenarios*, *storyboards* and *video prototypes*, which can either be animated or static, to present ideas of how a proposed system might be used in different ways, in different environments and situations, and with different people. These allow exploration of possible uses or the role a system should play in a person's life. The prototypes can either be passive, where they are merely presented to users and discussed after viewing, or they can be used as tools in participatory design (Sanders, 2002), where users are encouraged to contribute design ideas. In either case, however, the designer's role is to uncover underlying needs and values of the user, rather than to simply implement designs suggested by the users. Within a context of development for BCI users, the designer's ideas of how a system might be used

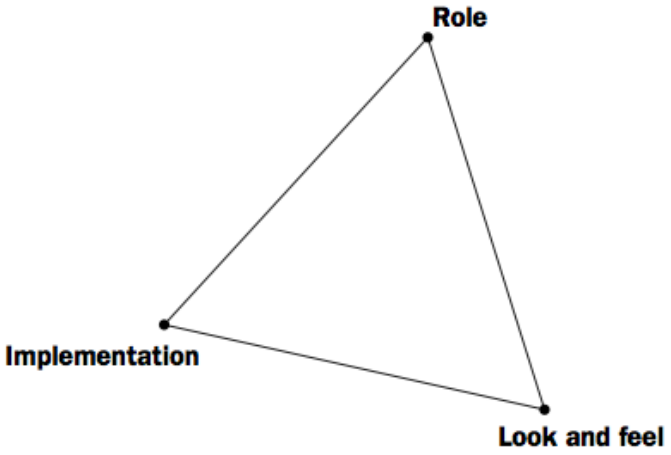


Figure 2.3: A model representing the functional space of prototypes. Taken from Houde and Hill (1997).

can be presented to end users in the form of stories. This concretises the designer's idea of what the user's environmental or social context might be like, such that the prototype is a discussion point for an end user to relate the context presented in the scenarios to their own. Their agreeing or disagreeing with the designer's idea of how the application might be used can then be grounded in a better understanding of the actual life and concerns of the user. As it takes time for the users to communicate, having a prototype as a discussion point saves time and energy as the user does not have to describe their situation from scratch, and even simple 'yes' or 'no' answers to questions can provide valuable insights for the designer to use.

Exploring the behaviour (implementation) of the system. Exploring how the system should work is an important part of design, as users need to understand the system (it should be usable) and find it pleasurable to use (it should create a good user experience). A mixture of low- and high-fidelity prototypes can be used to capture the behaviour of a proposed system, and have also been used in usability testing. With low-fidelity prototyping methods such as *paper prototyping* (Snyder, 2003), cardboard and paper cutouts can be used to represent buttons and menus on a graphical user interface. The participant to a user study can press fake 'buttons' on the 'screen', which the designer then modifies to simulate the system's response. This allows the designer to test how well people can understand and use a system. However, it is somewhat difficult to create interactive low-level prototypes for a BCI system for two reasons. Firstly, as a BCI is intended to be controlled by thought, it is not possible to simulate the input. This may be overcome by asking users to indicate verbally or by pointing to the correct controls. Since it is a prototype, it may be argued that this is not a matter of huge importance, and the method can be used to test that the basic logic of the system is understandable. However, if one wants to explore how the application should behave given the control characteristics of the BCI, it is tricky to do as the experimenter has to simulate the control, incorporating the noise and error of the

feedback. For this reason, the concept of video prototyping (Vertelney, 1989) may be more useful to present the behaviour of a BCI system to end users. Despite the passive nature of the method, it can be used to demonstrate the implications of a system having the characteristics of BCI, such as the length of time it might take to select an album in a music player, or the number of errors that might occur. This might be more useful in engaging end users to find out how they want the system to function given the characteristics of BCI.

Another prototyping technique is the Wizard of Oz (WOZ) technique (Bernsen et al., 1993), which is a method of simulating a system where its behaviour appears to be automatic, but behind it is an experimenter who is actually providing the system response based on the user's input. The name of the technique comes from the children's novel Alice's Adventures in Wonderland (Carroll, 1865), in which the seemingly magical ability of a powerful wizard from the land of Oz turns out to be no more than the actions of an elderly man hiding behind a screen. The technique is well-suited to investigating future systems that incorporate artificial intelligence in the interaction, such as in gesture and speech recognition. Aspects of the system can be controlled in order to investigate how potential properties of a system will be used and understood by the target end users. The participant can either be led to believe that the system is fully operational, or they can be informed of the existence of the wizard. In the latter, the wizard can either be visible to the user, or hidden behind a screen or in another room.

One property of the a system that can be controlled using a WOZ simulation is the uncertainty in the system. This makes it a prime technique to be used in BCI research, as a BCI system may be fairly unpredictable. For example, the performance of the system could be controlled such that one can compare usability aspects of a system, or user experiences of a system, where the control of the system is variable. Evidently, it is difficult to simulate a BCI system as the wizard has no way of knowing when an whether a participant is actually imagining the correct mental state. One way of getting around this, as with the low level prototype, might be to ask the user to point or verbally indicate their intended action. Such a means of prototyping not been explored in the BCI literature. It is possible that a combination of low- and high-fidelity prototyping can be used to explore the behaviour or usability of an intended design. For example, the interface could be a low-fidelity paper prototype, while the control of the BCI rendered by a digital simulator.

A more common means of investigating the low-level characteristics of the system is to use an automatic system and assume that the person is engaging with the task correctly. For example, Lynn et al. (2010) used a fake BCI to investigate the illusion of intention. Users were asked to try to move the cursor on the screen as much as possible, and it was found that a person's level of perceived intention to move the cursor could be manipulated by the number of times the cursor was perturbed. In this case, the system itself can actually be thought of as the 'wizard' modifying the user's input.

Other studies using online simulation have been used to investigate people's perceptions of systems with unreliable input. van de Laar (2011) investigated how the level of control

influences gamers' perception of fun. The experiment showed that for a simple game where users had to control the 2D position of a hamster, increasing control increases a sense of fun, up to a point where there is no challenge and it becomes boring. Plass-Oude Bos (2011) compared user experiences of using a 'utopia'-BCI and a 'real' BCI with respect to preferences for the use of different mental tasks to change aspects of a character in the game World of Warcraft. In the utopia BCI, users were asked to imagine the mental state and then press a button to indicate when they had completed the imagination. The authors found that ease of imagining the mental state was the largest factor for preference for mental states in the utopia BCI, whereas system recognition was by far the largest factor for real BCI. The example helps to understand how users expect an intelligent system to work. Cincotti et al. (2007a,b) used a noisy mouse input representing the noisy input of BCI to explore tactile feedback, showing that tactile feedback could compensate for visual feedback under high visual workload conditions. Other than these examples, simulation seems to be an under-used, but potentially highly valuable tool for BCI research.

In the examples above, general simulators have been used in order to capture a single aspect of a BCI system. However, as user performances can vary widely and there are control characteristics that affect the feel of control of a BCI other than noise and error, it would be useful to develop online simulators that capture the feel of control specific to individual BCI users. Such an online simulator would allow a designer or developer to uncover as many usability issues as possible that might be found when using a real BCI. Thus, a good online simulator should, as much as possible, capture the relevant control characteristics of the system that may affect the user's performance and experience when using a real BCI.

Experience prototyping and emphatic modelling. Experience prototyping, as introduced in Buchenau and Suri (2000), is defined as 'any kind of representation, in any medium, that is designed to understand, explore or communicate what it might be like to engage with the product, space or system we are designing'. Such prototypes are not intended to be used in formal usability testing, but at the design stage to engage with potential end users and to inspire designers to think about the potential issues and to come up with solutions. It can also be used to communicate the concept of the system to end users and stakeholders.

Empathic modelling (Poulson et al., 1996) can be thought of as a subset of experience prototyping, where the focus is on simulating disability in order to allow designers to understand a system from the user's point of view. This then enables the designers to appreciate the problems the system should tackle. Buchenau and Suri (2000) begins with a simulation where designers wore a pager which would beep intermittently, in order to experience the feeling of uncertainty and stress associated with wearing a chest-implanted automatic defibrillator which might deliver a shock to a patient at any moment. Through this method, the designers were able to appreciate the importance of providing a warning to patients and informing bystanders of what was happening. Other examples include the Third Age Suit Hitchcock et al. (2001), which is a wearable outfit which simulates the restricted movement experienced by the elderly for the purpose of design.

The online simulator developed over the course of this research can be used as an experience prototype as it aims to enable potential end users and stakeholders to understand what it feels like to use a BCI without actually using one. Both people who have never tried to use, or people who cannot use, a BCI can use the simulator to understand the control characteristics of the system, while for someone who is training to use a BCI, it may communicate what is possible with a BCI and thus motivate them to continue. In terms of empathic modelling, it might be used by someone who has perfect BCI control to experience the potential frustration involved in using a system with more control variability than they are used to. In the context of BCI, low-level simulation can potentially enable people to experience the frustration of using a BCI without needing to wear an EEG cap. On the other hand, learning to achieve control of a BCI can be fun and challenging, so in the same way that flight simulators are entertaining for people who might never become a pilot, understanding the skill required to control a BCI through using a simulator could bring respect for a trained BCI user.

Empathic modelling tools are typically used by designers and stakeholders to informally explore usability issues arising from the disability. Such tools are generally not used for usability testing with healthy users as a substitute for end users, or to make predictions about the actual performance of a user interface using a particular mode of interaction. One exception is the EASE tool which simulates the interaction of users with motor disabilities (Fait and Mankoff, 2003). Using this tool, it was found that adaptive word prediction is useful only for typing speeds less than 5-8 words per minute, correlating with previous findings in the literature. An online BCI simulator would be another exception to the rule, where it could be used to obtain qualitative feedback from users about the design of a particular control paradigm or application, as well as to predict task performance using either online or offline simulation.

2.4 Levels of simulation in BCI application development

The techniques used to develop a simulator or simulation model depend greatly on the purpose of the simulation. Simulation at different levels of the interaction process are used for different reasons and have different forms. In the current work, a distinction is made between *high level* simulation and *low level* simulation. In descending order, these techniques include:

- WOZ techniques and paper prototyping.
- Summary statistics of performance used for offline simulation. For MI-BCI, this includes the selection accuracy and speed.
- Simulation of the control signals. This can be used for simulating the feel of the interaction.

- Simulation of EEG. This has been used in BCI research for optimizing parameters for feature selection and classification.

2.5 A process for design and development

User-Centred Design (UCD) is a set of design guidelines that aim to ground the process of design in developing systems for the intended end users. 6 principles have been identified by the ISO 9241-210:2010 standard Human-centred design for interactive systems (Wikipedia):

1. The design is based upon an explicit understanding of users, tasks and environments.
2. Users are involved throughout design and development.
3. The design is driven and refined by user-centred evaluation.
4. The process is iterative.
5. The design addresses the whole user experience.
6. The design team includes multidisciplinary skills and perspectives.

Although UCD advocates involving the end user at all stages throughout the development process, in practice it is difficult to do this for participants who are locked-in or severely disabled. Ethical issues associated with asking users to carry out usability tests for long periods of time resulting in fatigue, or to test systems that are very much at the development stage, make it undesirable to consult the real end users at every stage of development. Thus the standard practice in developing ATs is to first test prototypes and systems with people with no motor disabilities before bringing them to end users. For example, in developing a breathalyser for a people with chronic obstructive pulmonary disease (COPD), Herriot (2012) found cultural probes, workshops and brainstorming sessions for eliciting user requirements to be unfruitful. The researchers found that the best way was to use video interviews lasting an hour, and even so this was effortful for the patients and would take a day to materialise. They then consulted medical personnel to understand the issues to be aware of, used empathic modelling to partially understand the difficulties one faces by using a breath restriction device, and used themselves as subjects in testing parts of proposed designs. Finally, a single 'super user' who was more committed to the study than the other 7 identified users evaluated the system. Thus requirements and needs may be elicited by end users, while development and some usability testing may be carried out by healthy users in the iterative stages of prototyping. Finally, the end users can be involved in testing and evaluating the final prototype.

Although such techniques have been described in AT and HCI literature, currently there does not appear to be a unified approach to developing applications for BCI for potential

end users. This is needed because the differences between end users needing to use a BCI and people who are able to use other modalities of input are large, both in terms of the constraints of current BCI technology and the abilities of end users. This leads to the main argument of the thesis for a role of simulation techniques in the process of developing BCI applications and novel paradigms. Offline simulations can be used to predict task performance and develop novel paradigms, while online simulations or prototypes at all levels can be used to getting the right design and the design right for end users (Buxton, 2007).

2.5.1 Proposed process for application design

In a proposed process for application design and development (Figure 2.4), end users would be involved mainly at the beginning and at the end of a version cycle of the application. End user requirements and expectations would be obtained through interviews and the presentation of video prototypes and storyboards. This would allow the designers to have a solid idea of the values the end user(s) have regarding the role and behaviour of the application, building a foundation of knowledge that the designers are 'getting the right design'. During the design stage, the use of simulators during design and initial evaluation with healthy users would contribute to 'getting the right design', before proceeding to the more expensive trials with real BCI. Naturally, if a new BCI control paradigm was being carried out, it would need to be developed with real BCI, but having an online simulator would allow for usability testing and longitudinal studies without needing to spend additional resources on real BCI.

The method can be thought of as a *mixed-users* approach where conclusions about user preferences, usability, core tasks and user requirements can be triangulated across users. Using a variety of research tools, with qualitative and quantitative data, with a larger range of users, can help to form a holistic picture of the requirements of end users leading to improved design. While quantitative data can be useful, large numbers of users must often be consulted for statistical analysis. As users may tire easily, the methods chosen for obtaining subjective feedback should be as effortless and enjoyable as possible. For example, if a user is unable to or finds it difficult to speak, quantitative methods can be used. If the end user can give feedback verbally, it may be easier and more fruitful for them to communicate their opinions in this way.

Figure 2.4 illustrates a process for designing, developing and evaluating BCI applications for end users, highlighting the roles that different prototyping and simulation techniques can be used. It should be noted that the design process is not a linear, waterfall model, but is iterative, and some of the stages may even be carried out in parallel. For example, designing a new paradigm to be used with the BCI for a specific purpose, in stages 3-5, may be begun in parallel with the user input stages 1-2, feeding in input from user feedback as and when it is available. The design process also begins with the assumption that the BCI paradigm is operable by the end user, since there is little point in developing an application

based on an input modality that the person cannot use. At each stage of the process, it is important for the designers to look beyond what the user says, uncovering their unspoken needs and values.

1. Identifying user requirements. Obtaining user requirements at this stage can be in the form of interviews, focus groups, questionnaires, or observations of the user in their environment. Simulation tools are probably not useful at this stage, although in a single session, this stage of the process may be mixed with stage 2 if the designers have some prior knowledge of the situation. The main participants in identifying user requirements would be the actual end users who have LiS, people who have less severe motor disabilities, caregivers, friends and family, and AT professionals.

2. Designing the role: getting the right design. In this stage, the designer presents potential scenarios in the form of storyboards or video prototypes as discussion points to find out the role that end users expect the proposed system to play in their lives. Again, in a single session the discussions and prototypes may blend with those in stage 3. The main user groups would be the same as those in stage 1.

3. Designing the look, feel and behaviour: getting the design right. In this stage, prototypes can be presented to users in order to establish how the prototypes should work. Video prototypes, Wizard of Oz simulations, and online (low level) simulators can be used at this stage, and as previously discussed, there can be a mixture of high and low fidelity simulations being used in a prototype. Offline simulations could be used to establish how long it might take to achieve tasks, and this might be communicated to the participants of this stage, who might be end users, and non-disabled people.

4. Implementation. At this stage, the offline simulations can be used to check the correctness of the system, again to time how long it would take to achieve tasks, while the online simulators would used by a developer to partially test network connections and how the system would work given the inputs from the simulator.

5. Evaluation with healthy users. During the initial evaluations, the online simulator is used for usability testing in conjunction with real BCI. The advantages of using the online simulator is that usability issues such as whether people can use and understand this system can be raised, leaving the BCI-specific usability issues to the real BCI. Offline simulations could be used to estimate how long a usability test might last.

6. Evaluation with end users. Finally, evaluation of a final prototype would be carried out with end users. The usability issues found with healthy users would be corrected such that at this stage, usability issues specific to the end users can be identified. Minor issues would be used to update the existing implementation, while feedback requiring major changes would be used as input into the next version of the system. Again, offline simulations could be used to estimate for a given user how long it might take to carry out a usability test. This will help in planning the tasks that can be

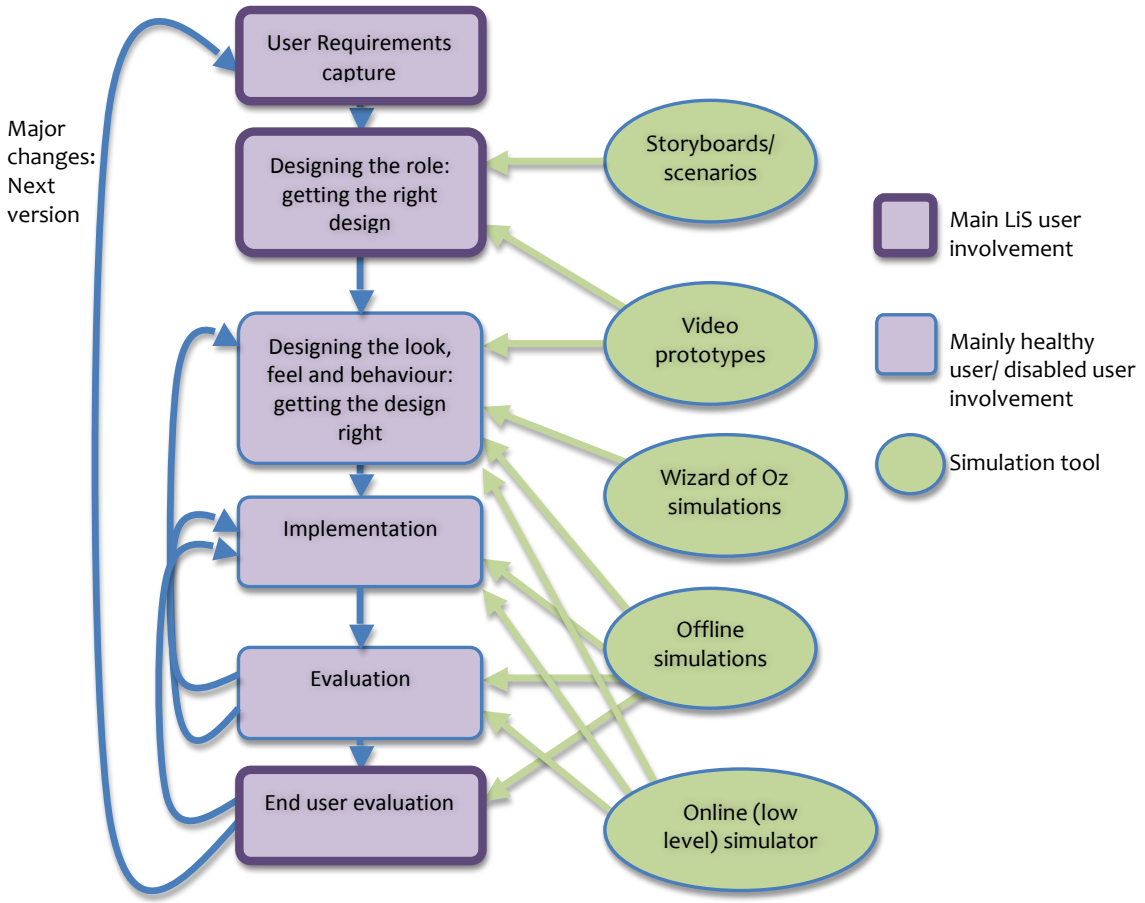


Figure 2.4: A design process for BCI applications incorporating the use of simulation techniques at different stages of the iterative process. End users are engaged at the beginning in finding out user requirements with the aim of 'getting the right design' Buxton (2007), and at the final evaluation of the system, while healthy and less severely disabled users are engaged in the design and evaluation of the system during the 'getting the design right' stages. The diagram shows where different simulation tools can be used in the design process. In general, low fidelity (or low *cost*) prototypes are used at the beginning stages of exploration, while the high fidelity tools are used in the later stages of implementation and usability testing. Video prototypes (Chapter 6), offline simulation (Chapters 3–5) and online simulations with the low level simulator (Chapters 4–6) are investigated in this thesis.

achieved in a given period of time, and ensuring that a participant would not be too fatigued to complete tasks.

As an example scenario for describing the design process, in the case of the lady who demanded to know why she was wearing such awful clothing (Kübler et al., 2001), a possible application would enable her to choose the clothes she wanted to wear. Some initial user requirements might be elicited from the end user by presenting her with several scenarios of how and when she would want to choose her clothing. If the characteristics of the user's BCI are known, the designer might choose to focus on the user's particular control characteristics. Offline simulations could be used to design the menu system by providing estimates of how long it might take to complete a task, for example selecting a set of clothing from the wardrobe. If known, the user's performance metrics would be used as input to the simulator. Possible selection mechanisms and menu hierarchies would be compared, and the best options for the user selected based on the simulation results. Low-level prototyping might be carried out with non-disabled or less severely disabled participants, followed by parallel development and usability testing with healthy users using online simulation and real BCI. For example, healthy users could test the system in an online simulation mode in order to establish how easily the interface can be understood. The combination of offline and online simulations with able-bodied and disabled people would allow the designer to obtain perspectives on different design options, eliminating as many usability issues as possible before the real BCI trials and presentation to an end user.

2.6 Conclusions

In designing applications for end users of a BCI who have severe physical disabilities, the requirements and abilities of end users must be considered along with the technical and interactional constraints of the technology. This chapter highlighted the control characteristics of an MI-BCI that make it difficult to apply rapid prototyping techniques for development. If customisation of ATs is important for end users in general, the level of tailoring to an individual must be greater still for BCI. It is argued that the development and use of models which pull together the different aspects of control can help to speed up design and development of BCI applications.

Specifically, offline simulation tools can be used to predict task performance. High- and low-fidelity online simulation tools can be used to engage both able-bodied and disabled participants in design and evaluation. These simulation tools can potentially be used in almost every stage of a UCD process, from design to implementation and evaluation. The exception to the rule is the user requirements capture stage, since at this stage the users' implicit and explicit needs are identified through interviews and other ethnographic tools. Further investigation into the value of the different simulation techniques are provided in later chapters.

3 Simulating Binary, Discrete Selection

Summary. Evaluations of BCI performance typically compare menu hierarchies and tree structures either in terms of the number of transitions required to reach a target, or the theoretical information transfer rate which incorporates the speed of selection. However, the time taken to achieve tasks may be a more useful and intuitive measure of task performance. This chapter describes how a model of selection accuracy and distribution of times for individual trials can be used to design menu hierarchies in finite state machines (FSMs) for MI-BCI systems. A tool for creating and evaluating different menu hierarchies is described and used to show that the time taken to achieve a task depends on the type of task, error correction method, selection accuracy of a single transition, as well as the time taken to make a single selection. In particular, it investigates menu hierarchies for tasks where there is a uniform probability of selecting an item in the menu, and whether or not an undo or delete is required for the application. Simulated performance of individual users in a spelling task is compared with online BCI trials.

3.1 Introduction

As described in Chapter 2, simulation and modelling can be useful in making predictions about user performance. This chapter focuses on offline simulation at the level of discrete, binary selections. The lower rate of communication possible with BCI due to the error rate, speed and limited degrees of freedom have prompted research into how best to optimize the communication rate of the end user. Several approaches to generating efficient binary menus with error correction have been described in the literature. Tregoubov (2005) computes the most efficient binary trees for which there are unequal probabilities of selection accuracies in the 2 classes. Bensch et al. (2007) used optimisation techniques to find the most effective finite state transducer for unequal classes, which was used both in a spelling application and a web browser. Dornhege et al. (2007) analytically derived the theoretical expected bitrate of various error correction/coding methods for 2-4 classes with selection accuracies less than 1.0. However, these applications do not take into account the implications of the time it takes to make individual selections, and how the time to make a decision (reach a leaf node) might be an interaction between the time it takes to make individual selections and the selection accuracy. The nature of a possible speed-accuracy trade-off itself is an aspect which has not been well investigated in the BCI literature: although there is usually implicit acknowledgement that some evidence accumulation is required to increase the recognition accuracy of the system, as far as the author is aware, there has not been any

formal investigation into how the selection accuracy changes as a function of time allocated for making a selection. In this chapter, simulation is used to explore the value of considering the time required for individual trials as well as the selection accuracy in designing menu hierarchies. For example, if a higher selection accuracy always leads to a better overall task performance, regardless of the time taken to make a decision, increasing the selection accuracy should remain the priority for improving the performance of an individual.

There have also been few comparisons with real BCI data to determine if the predicted measures actually compare with real data. This chapter thus presents an experiment comparing the performance of simulated users and their real performances. An approach to predicting task performance which simulates user input to menu hierarchies modelled as finite state machines (FSMs) is discussed. To begin with, the BCI used in the remainder of this thesis is described in the next section (Section 3.2), along with a description of the binary selection mechanism that is used in the calibration trials. In Section 3.3, measures of BCI task performance and simulation performance are described. Section 3.4 describes the use of FSMs to simulate task performance with an example of how to use the approach in Section 3.5, and Section 3.6 compares the predictions of simulated with actual data.

3.2 BCI used in the current work.

As mentioned in Section 1.5, the MI-BCI used in the course of this work was obtained from partners within the project. This was in part for convenience, as there would be no need to either develop the system from scratch or re-train users, as well as to explore how easily developers with no BCI expertise could interface with input from a BCI system. Although the choice of signal processing, feature extraction and classification methods are likely to affect the performance and control characteristics for a given user, regardless of the method used there is a range of individual user performances. Moreover, even within a given system the selected parameters and features are likely to affect performance. Thus, for the purposes of the simulation experiments of a discrete, 2-class system described in this chapter, the simulation methodology is likely to be valid for a range of MI-BCI systems of the same nature. Details of low-level characteristics, however, are likely to influence the methodology used to simulate the signals; this is discussed in detail in Chapter 4. In this section, details of the chosen BCI and the BCI experimental methodology for offline and online training are described.

The BCI system used was developed under Professor Millán at the École Polytechnique Fédérale de Lausanne (EPFL), the basis of which is described in detail in Millán et al. (2008), Galán et al. (2008) and Millán et al. (2004). Briefly, EEG signals obtained from 16 electrodes are attached to the scalp at positions FC3, FC1, FCZ, FC2, FC4, C3, C1, CZ, C2, C4, CP3, CP1, CPZ, CP2 and CP4 according to the 10-20 system, plus a ground electrode at FZ, and a reference electrode at the earlobe. A g.tec amplifier and data acquisition unit records the EEG at 512Hz, which is band-pass filtered between 0.1 Hz and 100 Hz; a Laplacian spatial filter is applied and the Welch power spectral density (PSD) is computed

at a rate of 16 Hz with a window size of 1s and overlap of 500 ms, over the 4-48 Hz band with 2 Hz resolution. Thus, a feature set consists of 16 electrodes $\times$ 23 bandpower frequencies = 368 features. Out of these, between 1 and 7 features are selected semi-automatically during the training stages, which saves computational power in online use. A Gaussian classifier is used to compute a probability distribution over the two mental classes, which is usually either left vs right hand, left hand vs feet, or right hand vs feet.

Prior to actual use, both the system and the user must be trained. The system is trained to recognise the user's mental states, while the user may require some time to understand the system and what mental activities lead to the production of the most stable features that can be separated by the classifier. *Offline* and *online* training can be distinguished. The online training trials are also called *calibration trials*, as they are used to indicate the baseline performance of a participant. In both training stages, a standard feedback paradigm is used. This enables the participant to select one out of n targets, where a target is a mental state or class that is reached by moving a feedback bar to the intended target. The number of targets corresponds to the number of mental classes for that system. For example, for a 2-class MI-BCI, there are two targets which can be left-right or top-bottom targets.

A *trial* dictates the flow of the interaction and indicates to the user at fixed periods of time which mental state should be performed. Figure 3.1 shows the sequence of a trial for a two-class MI-BCI. The trial begins with an inter-trial-interval, where the user takes the chance to relax and blink. A fixation cross is displayed which signals the user to mentally prepare to start performing the mental imagery. The target to be selected is then highlighted. This corresponds to the mental class a user should perform. The feedback begins moving as the person imagines a mental class. The trial ends when the feedback bar reaches either side of the bar; feedback is given to the user to reinforce the outcome of the trial: the target turns red if the incorrect target is hit and remains green if correct target is hit. A *run* consists of a set number of trials belonging to each class, presented in a randomised order. The work in this thesis has used 15 trials per class for each run.

The differences between the offline and online training trials involve the number of feedback classes that are shown, and the behaviour of the feedback. For a naive user (someone who has never used a MI-BCI before), the offline trials usually have three targets left, right and up, which correspond to imagining left hand movements, right hand movements or feet movements respectively. After the initial offline training, the two 'best' classes are selected. This can be a combination of user preferences (which imaginations felt subjectively the easiest or most comfortable to perform) and the classes that produce features that are the most easily distinguishable by the classifier. Since the target directions of the binary paradigm are fixed at left and right, the most natural mapping between the mental state and the target direction is used. For example, for the two classes left hand and feet, the left hand is assigned the left target while the feet are assigned the right target.

In offline training, the feedback bar automatically moves toward the correct target. In

online training (calibration trials), the feedback is updated according to the output of the user-parameterised classifier and the chosen integration technique, which defines the rate at which evidence is accumulated (see Chapter 4). The same selection mechanism can be used in an actual application where the selection of targets correspond, for example, to a 'yes' or 'no' decision, steps in a menu hierarchy, or intermediate nodes in a spelling tree.

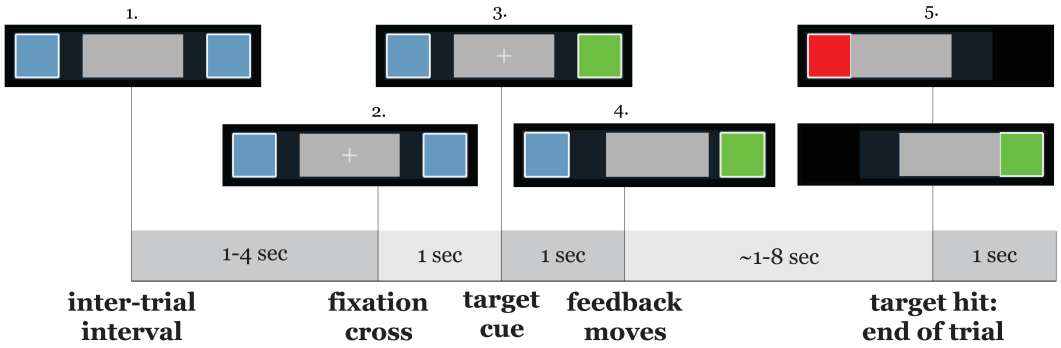


Figure 3.1: Timeline of a single feedback trial for offline or online training for a two-class MI-BCI, with screen shots of the *feedback paradigm* or *selection mechanism*. The flow begins at the inter-trial interval. The 'cursor', or *feedback bar* is stationary in the middle of the screen; the two blue squares on either side of the bar are targets which the user can select by imagining a motor-imagery (MI) class, which should move the feedback bar in the intended direction. After a variable time of 1–4s, a fixation cross appears in the middle of the bar. This signals to the user that the trial is about to start. Another second later, a system-selected target turns green to indicate that this is the one the user should try to select. The trial begins when the feedback moves, and ends when it hits a target. A trial is correct if the feedback bar has reached the marked target, and wrong if the other target is selected.

3.3 Measures

3.3.1 BCI performance measures

Several measures have been reported in the literature to compare the performance of individuals using a BCI. A set of guidelines for selecting the statistical measure to use for comparing classifiers and BCI systems in general can be found in Billinger et al. (2013). Some measures relevant to task performance in user interfaces are described below.

The *selection accuracy*, p, is defined as the proportion of trials correctly selected that the user intended to select. *Task* accuracy can be defined as the the proportion of tasks undertaken that the user was able to complete correctly. The selection accuracy for different classes may not be equal, such that the user may select one class more easily than another. A confusion matrix can be used to represent the individual selection accuracies of two or more classes, where the rows represent the intended selection class and the columns represent the

actual class selected. For example, a matrix $M = \left(\begin{smallmatrix} 0.7 & 0.3 \\ 0.0 & 1.0 \end{smallmatrix}\right)$, where the the cell M_{ij} represents the proportion of intended trials for class i actually selected as class j, represents a user with a p=0.7 for class 1 and 1.0 for class 2. The average value of the sum of diagonals is taken to be the user's overall selection accuracy. In the example, the average selection accuracy is $(0.7 + 1.0)/2 = 0.85$. In this case, there is a *bias* towards class 2, where the p=class 2 is significantly higher than for class 1 (where 'significantly' is a function of how the difference between selection accuracies affects the overall user performance, which is not investigated here. In general, however, equal selection accuracies in both classes is desirable for a binary menu hierarchy, while knowledge that a bias exists can be exploited, for example with the rotate-extend (REx) selection mechanism described in Chapters 5 and 6.). *Cohen's kappa* (κ) (Cohen (1960) via Billinger et al. (2013)) is commonly used as a measure of comparing the selection accuracy between systems for which there are more than two classes, or correspondingly, more than two targets in the control interface. The metric is useful for comparing performance across BCI systems which differ in the number of classes used since, given the same overall selection accuracy for two systems which have a different number of classes, the performance of the system with a larger number of classes is theoretically more accurate.

The speed of making individual selections, here referred to as the *time-to-selection* TTS, is a crucial measure of how quickly a task can be accomplished. In a MI-BCI, this is determined by several variables: preparation time for a user before the start of a trial, rest time after a trial, and the time taken to integrate the values of the classifier output over time, which accumulates evidence over time. In addition, the time taken to make a selection can be fixed, where the target closest to the accumulated evidence is selected after a specified period of time, or it can be variable, where the trial terminates when enough evidence has been acquired for the system to infer the user's intention. Since the geometric mean is a good approximation to the true median for reporting task times in HCI experiments for small numbers of trials (Sauro and Lewis, 2010), this value is used as a measure of central tendency for reporting task times for actual data.

Both p and TTS must be incorporated into an overall measure of performance. The *information transfer rate* (ITR), or *bit rate*, is commonly used as it incorporates both speed and accuracy, providing a measure of the maximum amount of information that can be transmitted through a channel. However, as Bianchi et al. (2007) explains, the actual user performance is dependent on the design of the control interface which uses the communication channel. Thus, the control interface must be optimally tuned to the transducer. The paper introduces the Efficiency Metric as a measure which compares interfaces with respect to the selection accuracy. However, it may be argued that, for a designer or stakeholder, such metrics are abstract and unintuitive. A useful metric to compare the combination of the control interface, selection accuracy and TTS could be the overall task time. Compared with identifying the number of selections or an abstract measure such as the bit rate, measuring the time taken to complete tasks is also a more intuitive measure for designers and stakeholders alike. Task time completion is also a standard metric used in HCI evaluations,

and some BCI evaluations of actual end user devices.

3.3.2 Simulation performance measures

Since the model inputs, at least for selection accuracy (if not time-to-selection), can be thought of as random variables, Monte Carlo simulations are used in this thesis to approximate the variables of interest (i.e. task performance variables). A measure of the average deviation of the simulated data from actual performance is given by

$$\frac{1}{n}\sum_{i=1}^{n}|y_i - x|,$$

where n is the number of simulation runs, y_i is a simulated variable and x is the corresponding recorded value from a real run. y_i and x can be single measures such as the time taken to spell a single word, or some measure of central tendency such as the mean, geometric mean or median time taken to spell a number of words. If the difference $y_i - x$ rather than the absolute difference is taken, this provides a measure of the tendency of the simulations to over- or under-estimate the actual task variable.

A *prediction interval* (PI) provides an estimate of how likely a given value produced by the output of a simulation is to fall within the specified interval. A prediction interval can be used to assess the level of confidence that one has in where the simulated variable of interest is likely to lie for a given instance of a real trial. In this thesis, a 95% prediction interval simply based on percentiles of the data is used, where, if α is the level of confidence one wishes to establish, then the PI is within the $[\alpha/2, 100 - (\alpha/2)]$ percentile range of the simulated data. The PI is distinct from the confidence interval (CI), which provides an interval within which one would expect to find the long run average value of the variable. Since the performance metrics obtained for each individual participant and each specified task is a single instance, for comparisons of simulated and actual data the PI is used as a measure of how well the simulator is able to predict the observed variable.

3.4 Modelling menu hierarchies using Finite State Machines

Menu hierarchies for binary selection may be represented using (deterministic) *finite state machines* ((D)FSMs). FSMs are widely used in Computer Science and Engineering disciplines for modelling and testing systems. They have been used in HCI to model simple systems such as mobile text entry (Sandes, 2005), calculators (Thimbleby, 2001) and other devices for predicting task usability. The advantages of FSMs are that they are simple to implement and can be used in a wide variety of applications. However, a drawback is that they are static and are mainly suitable for small state spaces. A formal definition of FSMs can be found in most introductory computing science texts, while applications of state machines to interaction design can be found in (Thimbleby, 2007). Briefly, a DFSM is a set of *states* (otherwise known as *nodes*, or *vertices*), which are connected by *transitions*

(or *edges*). States represent where the user is in a menu hierarchy or spelling tree and are graphically represented by circles with a corresponding state name. Transitions here are the detection of a mental class which is mapped to a interface selection class, such as left or right, and are shown with arrows. Terminating states are represented graphically by squares, where a selection(i.e. selecting a function in a menu hierarchy, or spelling a letter in a spelling tree) is made. The terminating states can be thought of as *letters* in an *alphabet* which can be used to spell a *word*. It should be noted that the letters can be menu items in a menu hierarchy, where a task is a 'word' which is achieved by selecting a particular sequence of menu items.

To simulate a task, a target state is first identified, and the current state reset to an initial state (root node) as defined by the structure of the hierarchy. At each state, Dijkstra's algorithm (Dijkstra, 1959) is used to find the shortest path (string of transitions) from the current state to the target state, until a terminating state is reached. The next target state is updated according to success or failure, and depending on the given task. The simulation for a single trial proceeds until a criterion is met by the simulation user. The inputs to the FSM are a user model which specifies the selection accuracy p and time-to-selection (TTS) for each transition. Selection accuracy is usually defined as the probability of making a correct selection, while the TTS can be either a distribution of times or a single value representing an expected time. Although for any given individual, there can be variability in the selection accuracy from run to run and session to session, this chapter makes a simplifying assumption that there is a constant selection accuracy for each user. Comparing simulated with real performance enables one to find out how useful the inherent stochasticity in the model is for predicting actual performance in the future. Inputs may be updated as a function of some observed behaviour such as the length of a task (for example, a user may become fatigued as time goes on, deteriorating the performance of the system); however, in the current models single inputs are used for each input variable. The outputs that are investigated are the number of selections made to complete a task, and overall time-to-task. Simulations were run using a set of python scripts which were developed specifically for this work.

3.5 Example: Comparison of task times for 4 menu hierarchies

In designing a binary menu structure, one can attempt to optimize the structure such that the fewest selections are needed to reach a target on average. If the probabilities of selecting different nodes are non-uniform (e.g. for a spelling task where some letters are known to be more frequently used than others), a new FSM can be recomputed as the posterior distribution changes given evidence of the user's intent as given by the input. For the menu hierarchies described in this section, a uniform probability of target nodes is assumed as this is a simple, generalisable case. Another consideration is whether an *undo* function is required. For example, in the case of a text entry system, a *backspace* is required to delete any wrong characters that are spelled. An example of a menu hierarchy which does not require an undo function might be in a music player, where if the wrong track is selected, one can simply try again and repeat the selection. Section 3.5.1 discusses recommendations

for where the user can simply *redo* or repeat the selection, while Section 3.5.2 discusses how requiring a correction (*undo*) affects the task time for different error correction mechanisms. In particular, the effect of the interaction between p and TTS with different menu structures is investigated.

With the assumption that a user is likely to make errors while browsing the menu hierarchy, an error correction mechanism embedded into the menu hierarchy may allow the person to reach the required node more easily. The menu hierarchy can be displayed such that from the user's point of view, the internal design of the hierarchy is transparent: the user simply needs to be able to quickly locate their target selection and identify the next mental state needed to produce to reach the target. Thus, in this analysis it is assumed that there are no extra cognitive effects of different menu hierarchies. A *back-up* transition allows the user to step back up the hierarchy upon realising that the wrong selection has been made. This has otherwise been referred to as a *confirmation tree* (Dornhege, 2006), where the option to go back up the tree can alternatively be viewed as confirming the user's previous transition. The redirection resulting from selecting the back-up transition can return the user to various positions in the tree. Here, the effects of returning the user to two positions up the tree, or back to the top of the tree, are investigated. The number of levels in the hierarchy after which a back-up transition is encountered must also be selected.

In addition, the possible benefits of a speed-accuracy trade-off are investigated in the next sections. Although it is generally acknowledged that both the p and TTS are important for providing information about a task, in MI-BCI research there has not been much research into how these co-vary. For example, Bensch et al. (2007) assumes that the usual time taken for selections is 5 seconds. The speed-accuracy trade-off is a well-known phenomenon in motor tasks where a person's accuracy in a task is traded off by the speed at which they carry it out. Thus, a higher error might be compensated by a faster selection rate while a higher selection accuracy might take a longer time; however this has not been explored in the BCI literature. It is possible that a higher overall task accuracy might be achieved if there is a lower p but a faster TTS. This section thus aims to provide estimates of how much faster task times need to be in order to maintain the overall task times if a lower selection accuracy is accepted.

As an example, 4 menu hierarchies with 8 letters are compared: (a) a simple binary tree without any back-up transitions, (b) a confirmation tree at level 2 (L2) returning the user to the top of the tree, (c) a confirmation tree at L2 and one at L3 returning the user to L2, and (d) a confirmation tree at L2 and one at L3 which transitions the user to an option to go either go back to the top of the tree or to L2. Examples of real applications that would use such a hierarchy might include selecting one out of eight applications to use, one out of eight browser links or typing a series of numbers (with a small increment to 10 letters in the menu alphabet).

The FSMs corresponding to each hierarchy are shown graphically in Figure 3.2. For each of the FSMs, a simulation run with 10000 trials were simulated for the selection accuracies

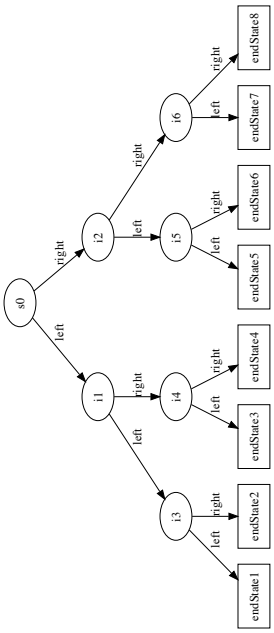
0.7, 0.8, 0.9 and 1.0. Simulations using the same number of trials have been reported in the literature (e.g. Müller-Putz et al. (2008)). Discrete values were selected for convenience and 0.7 was selected as the minimum selection accuracy required for reliable communication (see Section 2.2.4).

The analysis compares the performance of two task types: either redoing selections until they are selected, or requiring to select an 'undo' character. In the case of a *redo* task, selected letters are placed on the stack. The task is complete if the order of letters on the stack matches the order of letters in the target word. If an incorrect letter (a letter that is not the next target letter in the sequence) is selected, the stack is emptied. An FSM is considered infeasible for a particular selection accuracy if the number of times the stack is emptied reaches 100 (intuitively, a user might give up on their selection after being unable to attain it this number of times). For an *undo* task, the selected letters are similarly placed on the stack; however, upon selecting a wrong letter, the user has to select the letter designated as the 'undo' character in order to delete the wrong letter, before carrying on with the selections. The FSM is considered as infeasible if the number of letters on the stack reaches 15 more than the letters in the target word (preliminary experiments indicated that with this number of errors, it is difficult to make a recovery). The average number of selections (trials) required per bit is then calculated and compared with the theoretically optimal selections per trial.

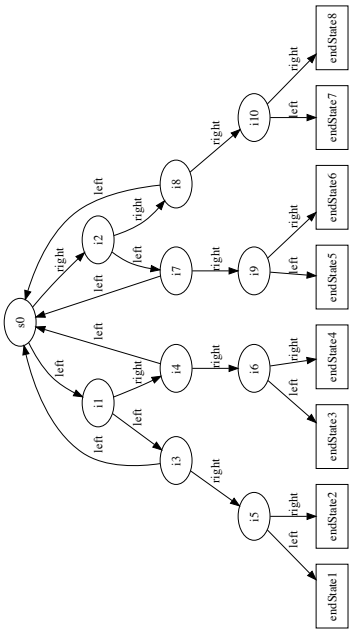
The time taken to complete tasks are compared with a potential speed-accuracy trade off where the range of times for making a selection is taken to be between 1-8 seconds. 1 second is a reasonable estimate for the minimum expected length of time required to make a single selection with a BCI, while 8 seconds was used for the maximum expected time as a BCI researcher suggested in personal communication that it is difficult for a person to continue imagining a limb movement for longer than this length of time. Thus for each menu hierarchy and task type, a simulation run was completed for each of the selection accuracies {0.7, 0.8,.. 1.0}, for each selection times {1, 2,.. 8} seconds, with the aim of investigating a possible interaction between the selection accuracy and TTS for the given menu hierarchies.

3.5.1 Binary menu hierarchies not requiring an undo or delete option

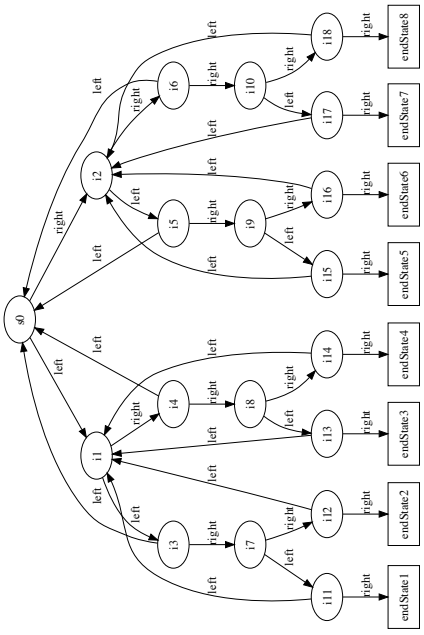
Effect of increasing number of letters on the required number of selections per bit. Figure 3.3 shows Box plots of the number of selections per bit of information over the selection accuracies (columns) and number of letters in a word (rows). As the number of selections required increases, the best FSM to use also changes. It can be seen that for p=0.7, a simple binary tree (FSM (a)) is the best option for selections requiring only one or two selections in terms of the average number of selections required. As the number of selections required increases, however, the FSMs with built-in error correction mechanisms allow the user to achieve the task within a lower number of selections per bit in comparison to the binary menu selection. For example, for selecting two consecutive letters, the average



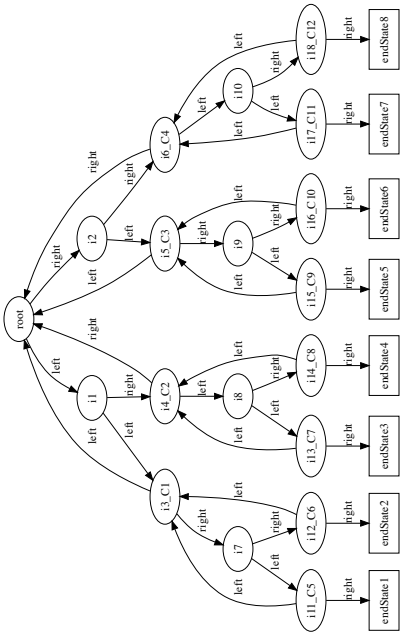
(a) Binary tree with no error correction



(b) Back-to-top at Level 2



(c) Back-to-top at Level 2, Back-to-L2 at L3



(d) Back-to-top at Level 2, Confirm L3

Figure 3.2: Menu hierarchies investigated in the chapter. Error correction mechanisms in the form of *back-up* transitions allow a user to transition back up the tree as soon as possible after realising that the wrong selection or transition has been made. Simulations are run to investigate where back-ups provide the best (if any) improvement in performance over the binary tree.

number of selections required for FSM (a) is 34.4, 31.8% fewer selections per bit than the theoretical minimum, while that for FSM (d) is 47.2, 6.7% less than the theoretical minimum. However, to get 4 consecutive letters the average number of selections required for FSM (a) is 323.7, 220.0% more selections per bits than the theoretical minimum, and for FSM (d) it is an average of 146.1 selections and 44.6% over the theoretical minimum.

An interesting result is that one FSM may 'win' on the average number of selections required, but have a lower variance and lower expected maximum than another FSM. For example, for a p of 0.9, the average number of selections per bit for spelling 5 letters is 2.8 for FSM (a), and the 95% PI range is (1.0–8.8), while for FSM (d) the average is 2.4 with a PI range of (1.7–4.1). Thus, while the average number of selections needed is somewhat lower for a simpler FSM (a), the expected upper range is more than twice that needed for the more complex FSM (d) with error corrections within the menu hierarchy. For the FSMs considered here, the phenomenon does not occur for selection accuracies of 0.8 or 0.7, where the trend of the average and variance always correspond to one another relative to different FSMs.

Speed-accuracy trade off. The speed-accuracy trade off is examined in Figure 3.4, where the purpose is to find out the extent to which trading off a higher p for a longer TTS (making decisions more certain at the cost of taking a longer time to accumulate evidence) can increase the time taken to select words. For selecting a small number of letters in the redo task, a lower selection accuracy but a quicker selection rate decreases the time it takes to spell the word. For example, a p of 0.7 with an average 3 second TTS is likely to take between 9 and 81 seconds, while a p=0.8 with an average TTS of 5s is likely to take between 15 and 90s. For two letters, a p=0.7 and TTS of 3 seconds (mean 103.3s, PI range 18-342s) is an improvement over a p=0.8 and TTS 7s (mean 160.3s, PI range 70-378) in terms of both the average and the variance. If the TTS for p=0.7 can be reduced to 2 seconds (mean 68.9s, PI range 12-228s), this is better than for 0.8 accuracy at 5 seconds (mean 114.5s, PI range 50-270s).

As the number of letters per word increases, a trade-off can only be exploited if the selections for the lower selection accuracy reaches 1 second. For example, at 5 letters per word, a p=0.7 and TTS of 1s has an overall lower task time (mean 227.2s, PI range 46.0-726.0s) than a p of 0.8 and TTS=5s (mean 340.1s, PI range 155.0-800.0s). For a p=0.8 and TTS 4s, the mean task time is longer at 272.1s; however, the PI range is 124.0-640.0s: the upper bound on the task time is lower than for p=0.7. On the other hand, if TTS for a p=0.8 can be reduced to one second, this reduces the expected time-to-task to 62% of the expected task time for selection accuracy 0.9 and 3 second TTS (mean 108.7s to 68s; PI range 75.0-183.0s to 31.0-160.0s).

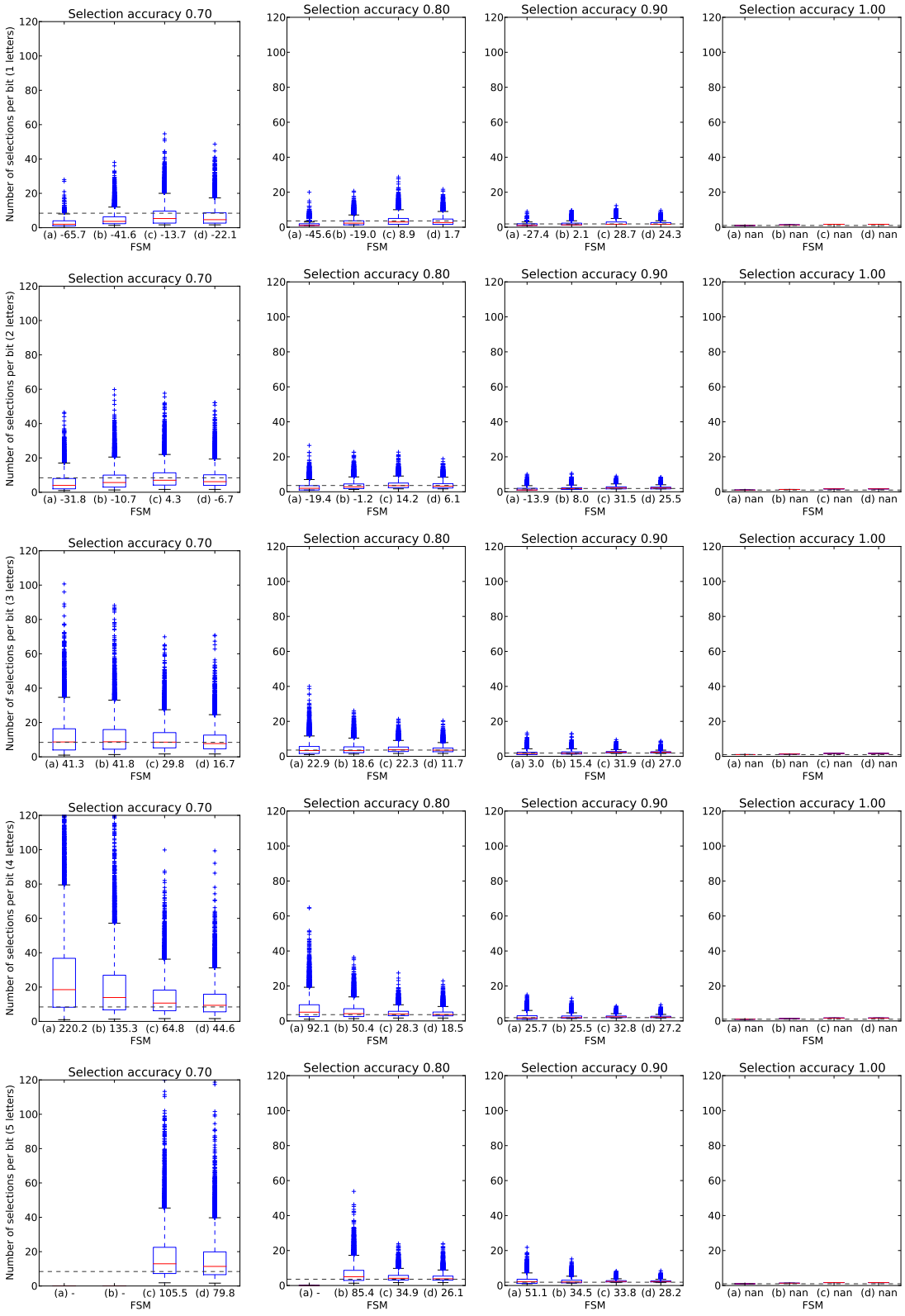


Figure 3.3: Box plots of the number of selections per bit required to select letters from an alphabet where the task is simply to *redo* the selection if the wrong letter is selected (lower values are better). The rows represent the number of letters requiring to be selected in sequence (1-5), while the columns are the selection accuracies 0.7, 0.8, 0.9 and 1.0. Box plots correspond to the FSMs (a)–(d) in Figure 3.2. An absence of a Box plot indicates that the task was not achievable for the particular selection accuracy-FSM combination. The percentage number of selections per bit more than the theoretical minimum required for a binary symmetrical channel is displayed for each FSM on the x-axis label.

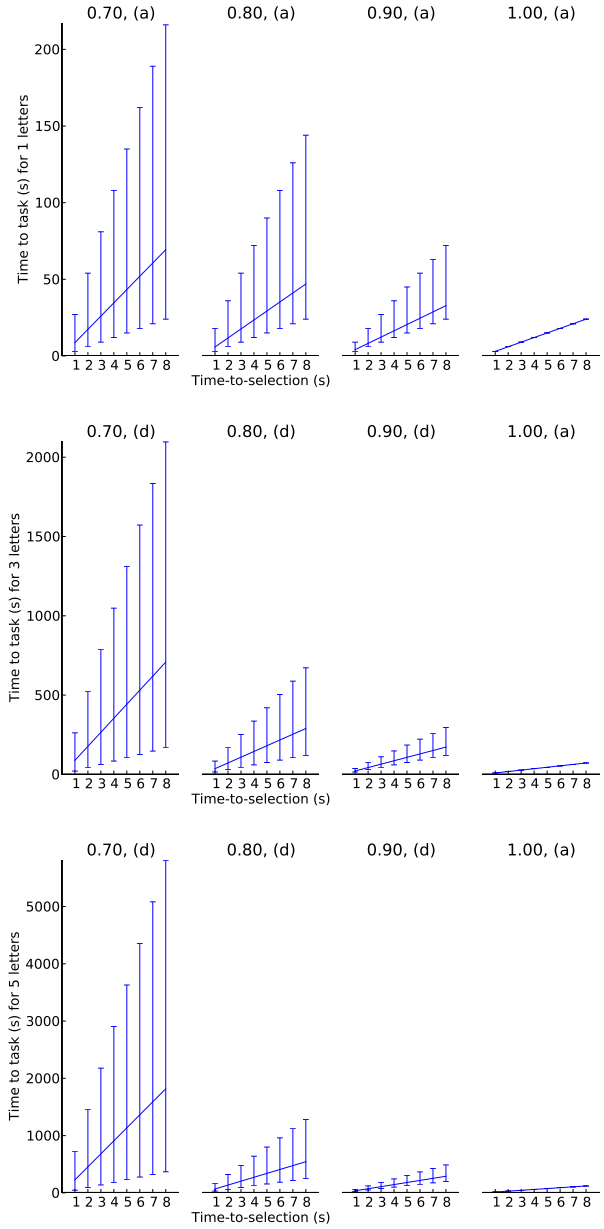


Figure 3.4: Comparison (mean and 95% percentile range) of selection accuracies p (columns) and time-to-selection (TTS) on selecting 1, 3 and 5 letters out of 8 (rows) for 4 FSMs, for *redoing* letters if an incorrect letter is selected. Values are shown for the best FSM for each selection accuracy, where the best FSM is the one with the lowest upper PI bound on the number of selections per bit.

3.5.2 Binary menu hierarchies requiring an undo or delete option

Effect of increasing number of letters on the number of selections required per bit of information. The distribution of the number of selections per bit required to spell words of 1-5 letters are shown for each of the FSMs in Figure 3.5. It can be seen that if only the average number of selections per bit are taken into account, different interfaces are the best for different selection accuracies. The best FSM is the same regardless of the number of letters in a word, but the average number of selections per bit required to complete the task (spell a word) increases as the number of selections increases. Based on averages, FSM (d) is the best for selection accuracies of 0.7 and 0.8, while FSM (a) is best for selection accuracies 0.9 and 1.0. for p=0.7 and selecting 2 letters, the task is impossible with a simple binary tree (FSM (a)), but requires 31.5% more selections per bit than the theoretical minimum, with an average of 62.2 selections. Selecting 4 letters requires an average of 130.9 selections, which is 38.4% selections per bit more than the theoretical minimum of 8.4 selections per bit.

Again, for p=0.9 it is found that although the average number of bits per selection is lower for FSM (a) than for FSM (d), the 95% PI has a larger range for FSM (a) than for FSM (d); this corresponds with a lower expected upper bound on the number of selections per bit for FSM (d). The average number of selections required per bit increases from 2.1 for selecting one letter to 2.3 for selecting a word with 5 letters using FSM (a). However, the range decreases from (1.1-9.6) to (1.1-5.6). For FSM (d), 2.5 selections per bit are required for words with 1-5 letters. Again, the range decreases from (1.8-5.7) to (1.8-3.9). Interestingly, for all the FSM-selection accuracy combinations, the average number of selections per bit increases as the number of letters in a word increases but the upper bound decreases.

Speed-accuracy trade-off. Figure 3.6 compares the effect of 1–8 second TTS on the expected times required to spell words that are 1–5 letters long. It can be seen again that in comparing the task times, both the average and range of times should be taken into account. For example, for spelling one letter with a p=0.7 and 2s TTS, the average expected time is lower than for p=0.8 and TTS=5s (mean 56s compared with 57.2s), but the upper bound on the expected 95% PI range is higher (10.0-268.0s compared with 25.0-175.0s). To do better than the expected range, the TTS for p=0.8 must be around 8s (mean 91.5s, PI range 40.0-280.0s). At 2 seconds, a selection accuracy of 0.9 is always better than a selection accuracy of 0.7.

Spelling one letter with a p=0.7 and 1s TTS (mean 28s, PI range 5.0-134.0s) allows a lower expected task time over a p=0.8 and 4s TTS (mean 45.7s, PI range 20.0-140.0s). This is an average time saving of 39%. A similar trend is found for selecting 5 letters (mean 164.9s, PI range 46.0-451.0s for p=0.7 and mean 239.9, PI range 124.0-456.0s for p=0.8, 32% time savings), showing that there is value in trading off a faster TTS for a lower p at p=0.8. However, the average time taken to spell a 5-letter word with p=0.7 at 1 second per selection (164.9s) is less than that for a p=0.9 at 5s per selection (177.8s), the PI range is considerably larger (46-451s compared with 125-275s). Even at a TTS of 8s, the selection accuracy of

0.9 provides a lower expected range of task times (mean 284.6s, PI range 200.0-440.0s). However, here the expected time for p=0.7 is 42% less than p=0.9.

For spelling a 5 letter word, going from p=0.8 to p=0.9 is better when a 2s TTS for p=0.8 (mean 120s, PI range 62.0-228.0s) is compared with a 5s TTS for p=0.9 (mean 177.8s, PI range 125.0-275.0s). If the time can be reduced further to 1s (mean 60s, PI range 31.0-114.0s), this is better than a p of 0.9 at 3 s (mean 106.7s, PI range 75.0-165.0s).

3.5.3 Discussion

This section demonstrated how task performance in terms of the number of selections and time required could be estimated using FSMs to represent menu hierarchies, and a model of the user represented by the selection accuracy (p) and time taken to make a single selection (*time-to-selection*, TTS). The performance metrics considered were the number of selections [per bit of information] and time taken to complete tasks, and the independent variables apart from those in the user model were the task type, *redo* or *undo*, and the number of letters in a word.

It was shown that the best FSM to use for a given p depends on the nature of the task. For the redo task, the best FSM for selecting one or two consecutive letters or menu items is the binary tree without any built-in error correction mechanism. However, as the number of consecutive letters to be spelled in succession increases, the best FSM to use may be one that requires more opportunities for correcting mistakes before reaching the end of the tree. For the undo task, the choice of FSM remains the same for increasing numbers of letters in a word. (It is important to note that for the redo task, the number of selections per bit is less than the theoretical minimum because it does not provide the same amount of information as the undo task; nevertheless the values are reported for comparison.) The finding suggests that in selecting an FSM to use, the nature of the task should also be taken into consideration. To optimise the task performance for a given application, it may pay the designer to think carefully about what the application should achieve and how the user will wish to use it. For example, in choosing a paintbrush in a menu, if the wrong one is selected it is possible to just select it again; in this case a binary menu can be used. However, on applying the paintbrush the user will likely wish to undo or delete mistakes; in this case an FSM with back-up options will be more efficient. The example also highlights the difference between tasks where future 'letters' selected either have or do not have an impact on previous letters selected, and between 'letters' (most likely menu options) that have an immediate effect compared with tasks that are a cumulative product of several letters (such as in a spelling task).

As the number of letters to be spelled increases, the percentage number of selections required per bit increases relative to the theoretical minimum. This effect is larger in the redo task and with lower selection accuracies. It was also found that an FSM having a lower expected performance may in fact have a larger range, corresponding to a higher upper bound on the

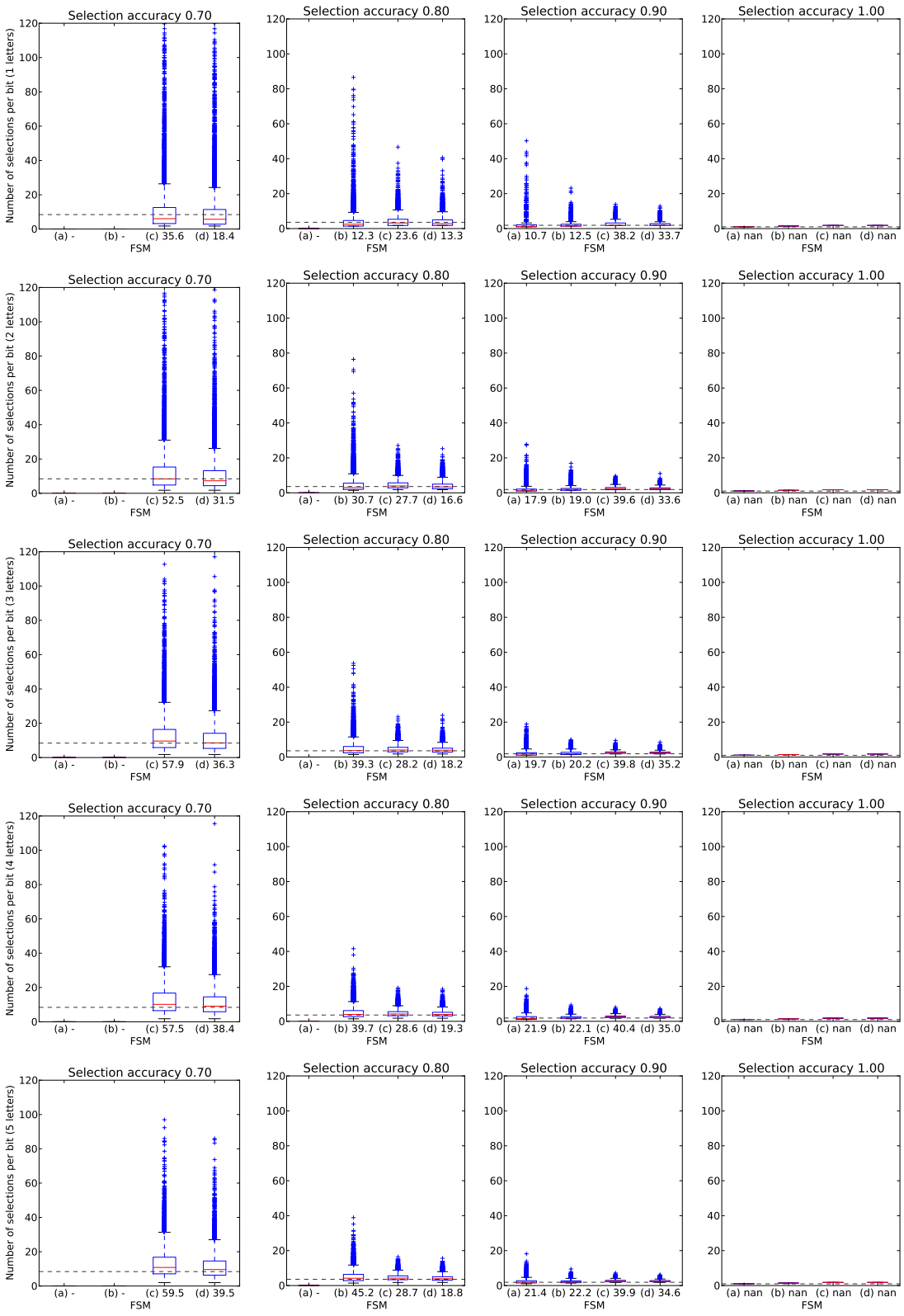


Figure 3.5: Box plots of the number of selections per bit required to select letters from a 7-letter alphabet where the task is to select an 'undo' option if the wrong letter is selected. Lower values are better. The rows represent the number of consecutive letters requiring to be selected (1-5), while the columns are the selection accuracies 0.7, 0.8, 0.9 and 1.0. Box plots correspond to the FSMs (a)–(d) in Figure 3.2. An absence of a Box plot indicates that the task was not achievable for the particular selection accuracy-FSM combination. The percentage number of selections per bit more than the theoretical minimum required for a binary symmetrical channel is displayed for each FSM on the x-axis label.

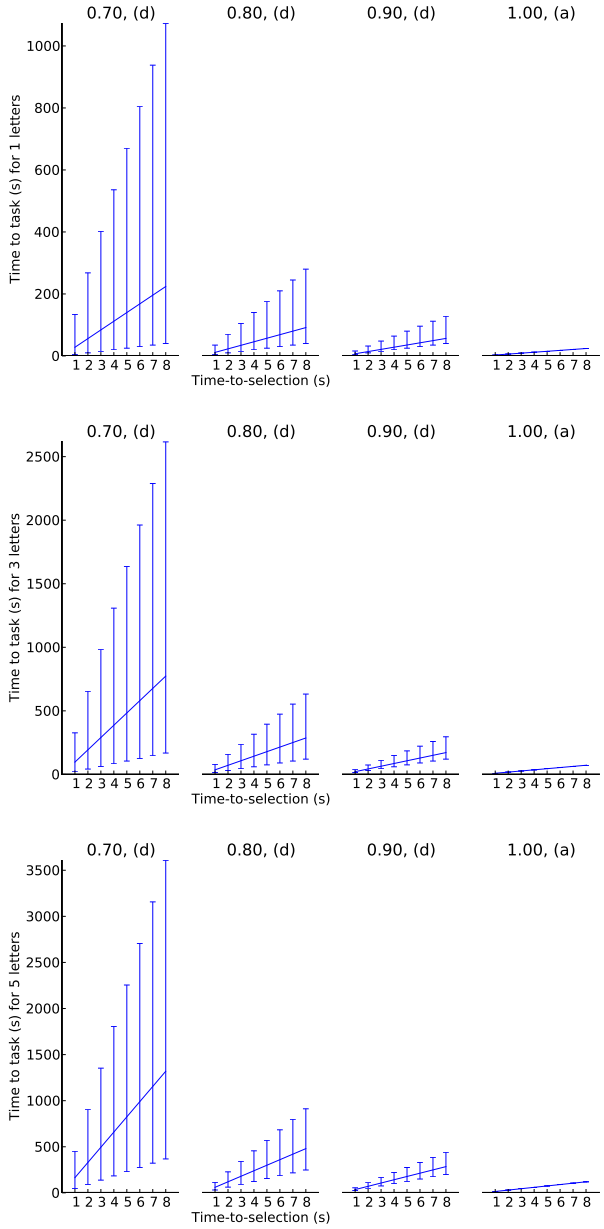


Figure 3.6: Comparison (mean and 95% percentile range) of selection accuracies (p, columns) and time-to-selection (TTS) on selecting 1, 3 and 5 letters (rows) out of 7 for 4 FSMs, for selecting an *undo* option if the wrong letters are selected. Values are shown for the best FSM for each selection accuracy, where the best FSM is the one with the lowest upper PI bound on the number of selections per bit.

95% PI, than an FSM that has a higher expected performance. It is important to note that in designing a user interface with the goal of achieving the best possible user experience, the user's *perception* of task performance is arguably more important than the actual task performance. Accordingly, it is important to identify the criterion that a user would use to decide which interfaces are the most pleasurable (or least frustrating). Future work could thus seek to address whether a user's perception depends on the expected (mean or median), mode, lower bound or upper bound of the actual time taken, as this should determine the metrics the designer uses to compare and optimise FSMs.

The trade-off between speed and accuracy was investigated, showing that in certain situations there can be a performance benefit to increasing the selection accuracy at the expense of increasing the TTS for a single trial, and vice-versa. It was seen that, if the time-to-selection is made considerably lower than has currently been achieved with an MI-BCI, a lower selection accuracy could match, and even outperform, the time required to complete tasks. In reality, this is dependent on at least two factors. Firstly, the extent to which a speed-accuracy trade off can be observed in MI-BCI remains to be explored. Secondly, whether or not a user can be trained to use an MI-BCI operating at such a speed remains to be determined. Such a user is likely to be an expert at least in determining the correct selection to make, even if they are not an expert at the motor imagery task itself.

Although a speed-accuracy trade-off exists wherever there is motor movement, it is possible that no such trade-off exists within MI-BCI. This may be because, unlike in a reaching task as with motor movement, the ability of the participant to produce the correct mental state is an either-or situation. Thus, people who are good at the MI task can already produce the required mental state quickly, and for these users there is no benefit to decreasing the TTS to have a selection accuracy that is much worse. On the other hand, for people who are not as good at the MI task, it is possible that a faster TTS at the expense of a lower p may in fact be better than a slower TTS at the expense of a higher p, if the TTS can be reduced to as little as 2s including the preparation time between trials. The findings also show the benefits of improving the selection accuracy of the user even with a longer TTS if the minimum trial time cannot be reduced sufficiently.

Thus, trade-offs have been observed in terms of the overall task completion time such that there are interactions between the TTS, selection accuracy and nature of the task. Informal interviews conducted with participants evaluating a scanning interface (Bhattacharya et al., 2008) suggest that a higher selection accuracy with a longer TTS is preferable to a lower selection accuracy with a faster TTS. Future work could aim to find out the extent to which this applies to BCI control. It would also be useful to find out whether the user's perception of the time taken to achieve tasks depends on the selection accuracy and time of single trials, or whether it corresponds more to the overall ability to achieve a task and how long the task takes.

3.6 Validation with real data

The results from any simulation of a human-computer interface must be taken with caution, as human factors that had previously not been considered may surface. In BCI, this is particularly true as the field is relatively new and much research into the psychology of brain-computer interaction must exist. Comparisons of model predictions with real data can either help to strengthen and validate the models, or they can highlight weaknesses in the model for improvement. Either way, obtaining a theoretical prediction of the task performance for a particular user is useful. In this experiment, data from standard binary calibration trials were used to predict the performance of a spelling application.

3.6.1 Method

The simulation task was a binary speller which did not use a language model (i.e. there was uniform probability for selection of each letter, Figure 3.7). Although there was an 'undo' option built into the tree, a step back function using electromyography (EMG) was used instead to step back up the tree, essentially undoing the last command. 6 participants (3 healthy, 3 disabled participants - participant code prefixed 'pXX') who had previously been trained to use a motor imagery BCI attempted to spell the words 'hello', 'email', 'computer' and 'internet'. As there were no calibration runs carried out on the days of the experiment, calibration data prior to the spelling experiment was obtained from the experimenter[1] who designed and conducted the experiment. For each participant, 2 or 3 sessions of calibration runs prior to the experiment were made available.

For each of the participants, the performance from the calibration runs were used to simulate the spelling task. Namely, these were the selection accuracy and the time-to-selection (time taken to select a single trial). Two different ways of using the data to simulate the spelling task were used. In the first case, the average p over all the runs, and the geometric mean of the TTS across all trials for the last session prior to the experiment was used. If the performance of the first run appeared to be significantly poorer than the other trials, it was discarded as this would be taken to be a practice run. Each word was simulated 500 times. In the second case, each word was simulated 500 times for each run across the last 2 sessions (similarly discarding practice runs). The recorded metrics were the number of selections (including undos) and the average time taken to spell a word or a character. 500 runs were used as the means and variances were found to be fairly constant at this number of runs. The average selection accuracy was taken over the runs for the last session as this would lessen the effect of any outliers to achieve a more precise prediction, while the individual runs from the larger sample were used in order to obtain a wider prediction interval, which would strengthen the confidence in the simulation.

Model assumptions. As the undo was due to an EMG signal which allowed the user to undo the last command, this was taken to be fixed at 4 seconds (3 seconds freeze time + 1 second)

[1] Thanks to Serafeim Perdikis for the data.

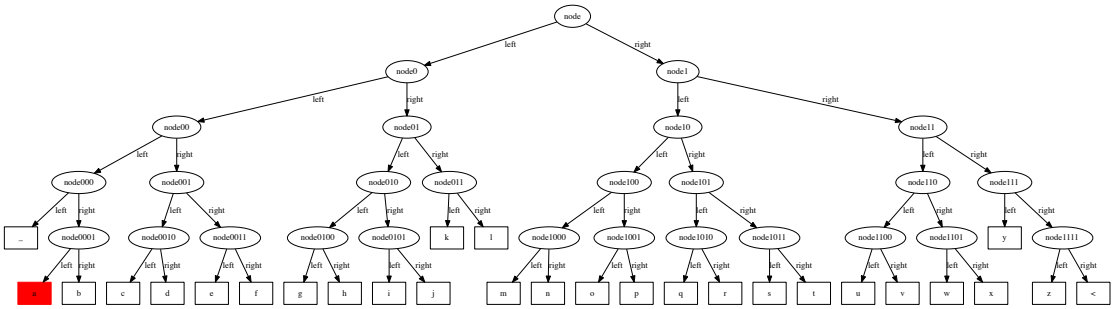


Figure 3.7: Finite State Machine (FSM) for the text entry simulation with no language model. The system is essentially a binary tree.

as a reasonable estimate and confirming this with the average undo times for 4 users. It was assumed that the user never made mistakes with the undo command, such that there were no false positives, and the user could correct themselves every time a mistake was made. Since the undo command is used to undo the last BCI command, if an undo command was selected when the state of the FSM was in the root node, the system reverted to the state before a character was selected. In other words, every time a transition was made, the previous state was saved in the FSM.

3.6.2 Results

Calibration runs. Figure 3.8 shows examples of calibration runs for 3 participants, a7 (healthy), pAL (end user) and pSI (end user). It can be seen that in general there are slight fluctuations in the task accuracy. Accuracy for end user pAL improved as the number of sessions went on. However, for pSI, the performance of the last session was poor, albeit increasing to a mean accuracy of 0.8 for the last run.

One finding of relevance to the interpretation of the simulation results is that the values for the integration and threshold from the calibration session were sometimes different from that of the experimental session. These are parameters which affect the time taken to make a selection, and in some cases the selection accuracy (which is why optimisation to individual performance is necessary). To partially correct for this, a constant value was subtracted from the selection time for each trial where there was a difference between the integration rate and thresholds. This was calculated as the difference between the minimum time it would take to make a selection for the integration and threshold values for the calibration and experimental sessions.

Word predictions. Figure 3.9 compares actual and predicted performance of the spelling task. In general, the trend of the expected means are in line with the task times, as the mean absolute error (MAE) taken to spell a word was 1.37min across the simulations from data for the last session, and 1.70min for the simulations from data from 2-3 previous sessions

(Table 3.2). The maximum underestimation was -6.31min for participant pLI, spelling the word 'internet' (actual time 11.9min), and the maximum overestimation was 4.02min for participant pPAL spelling the word 'computer' (actual time 7.02min). The mean number of selections was within -47 and +41 selections for participant pSI spelling the words 'hello' and 'email' (actual 93 and 28 selections), while the MAE was 10.77 (data from last session) and 9.34 (data from 2-3 previous sessions, Table 3.1). While the number of iterations was generally slightly underestimated (-2.51 and -3.53 iterations from the mean for last and previous sessions), the time taken was generally overestimated (5.58 and 32.44 seconds). Although the expected means estimated from the last session were, on average, closer to the actual time than that expected for the model based on data over a longer period of time, the prediction intervals contained the actual value 95.7% (22/23 words) of the time for estimates based on 2-3 previous sessions as compared to 39.1% (9/23 words) for the estimates for the last session. A similar result was found with the estimates for the number of selections (91.3% for previous sessions and 34.8% for the last session). For comparison, the prediction intervals from taking individual runs for the last session contained 60% of the actual values for both number of selections and time to spell a word, and prediction intervals from data based on the average performance for each of the previous sessions contained 87.0% and 60% of actual values.

Character predictions. From Table 3.3, it can be seen that overall, the average number of selections per character predicted was fairly close to the real trials, within 1.4 selections from the actual as predicted from the last session, and 1.7 selections for the runs from previous sessions. The variance of the time to select a character (Table 3.4) was higher in proportion to the number of selections, overestimating by as much as 12.6s in the predictions from the last session and 30.9s in those including a larger number of runs. In general, there was a slight underestimation of the number of selections and an overestimation of the time required, although this should be taken with caution as the sample size is small. Although the MAE was lower for predictions based on trials for the last session than those based on a larger number of sessions, the prediction intervals for the latter captured the real averages.

Table 3.1: Comparison of the mean predicted and mean actual number of selections over words. 'Diff' is the $Mean\ predicted - Mean\ actual$ and 'Abs Diff' is the $Mean\ abs(predicted - Mean\ actual)$.

Word	Actual	Last session			2-3 Prev sessions		
		Mean	Diff	Abs Diff	Mean	Diff	Abs Diff
hello	36.50	30.53	-5.97	8.25	29.51	-6.99	10.36
email	28.33	33.81	5.48	10.39	31.66	3.33	7.73
computer	56.50	54.44	-2.06	11.06	51.98	-4.52	7.75
internet	51.60	44.12	-7.48	13.40	45.65	-5.95	11.53
Means	43.23	40.73	-2.51	10.77	39.70	-3.53	9.34

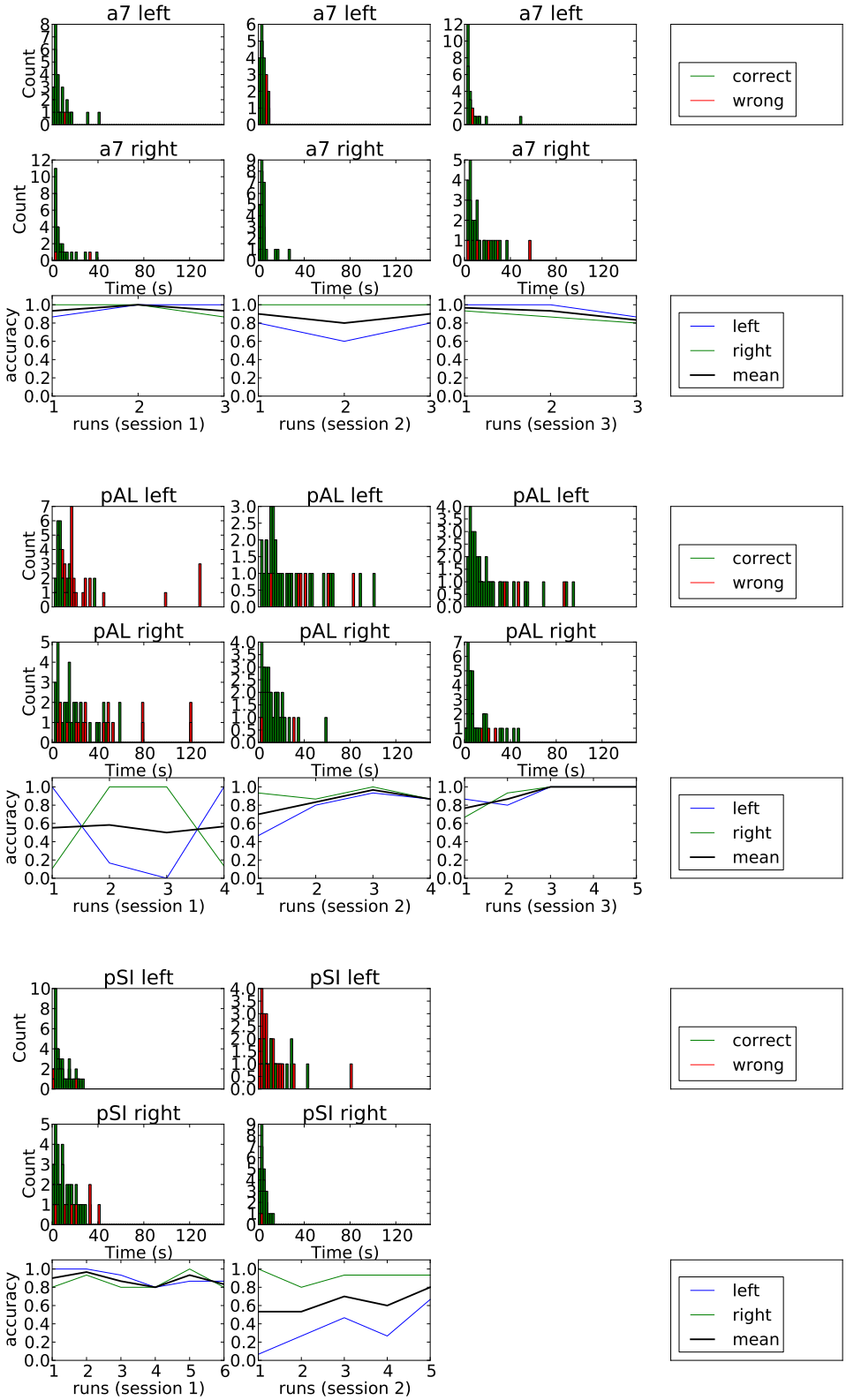


Figure 3.8: Histogram of time to selection (top 2 rows) and accuracies (bottom row) over last runs prior to experiment for 3 participants a7 (non-disabled), pAL and pSI (disabled). In each case, the average selection accuracy and time for correct trials for the *last session*, and the individual runs over the last *2-3 sessions* were used for simulating the spelling application. For pAL and pSI, the first run from the sessions were omitted from use.

Table 3.2: Comparison of the mean predicted and mean actual timing (seconds) over words. 'Diff' is the $Mean\ predicted - Mean\ actual$ and 'Abs Diff' is the $Mean\ abs(predicted - Mean\ actual)$.

Word	Actual	Last session			2-3 Prev sessions		
		Mean	Diff	Abs Diff	Mean	Diff	Abs Diff
hello	222.50	243.77	21.27	29.63	261.25	38.75	53.06
email	230.83	275.34	44.51	61.85	287.26	56.43	63.98
computer	458.00	439.42	-18.58	75.96	466.19	8.19	137.58
internet	396.00	371.12	-24.88	152.12	422.39	26.39	129.81
Means	326.83	332.41	5.58	79.89	359.27	32.44	96.11

Table 3.3: Average number of selections per character for actual and simulated trials. *Diff* is the $simulated - actual$ number of selections.

Participant	Actual (±SD)	Last session		2-3 Prev sessions	
		Mean (95% PI)	Diff	Mean (95% PI)	Diff
a7	4.8 (0.2)	5.8 (4.6, 7.0)	0.9	6.0 (4.6, 9.2)	1.2
b2	6.2 (0.8)	4.9 (4.6, 5.0)	-1.4	5.1 (4.6, 6.5)	-1.1
e7	5.6 (0.7)	5.8 (4.6, 7.2)	0.1	5.5 (4.6, 9.0)	-0.2
pAL	6.4 (1.5)	5.2 (4.6, 6.0)	-1.2	5.6 (4.6, 8.0)	-0.8
pLI	6.1 (1.4)	5.2 (4.6, 6.0)	-0.9	5.5 (4.6, 8.2)	-0.6
pSI	11.7 (5.3)	12.7 (8.2, 18.4)	1.0	10.0 (4.8, 25.2)	-1.7
Mean	6.80	6.58	-0.22 (MAE 0.92)	6.28	-0.52 (MAE 0.91)

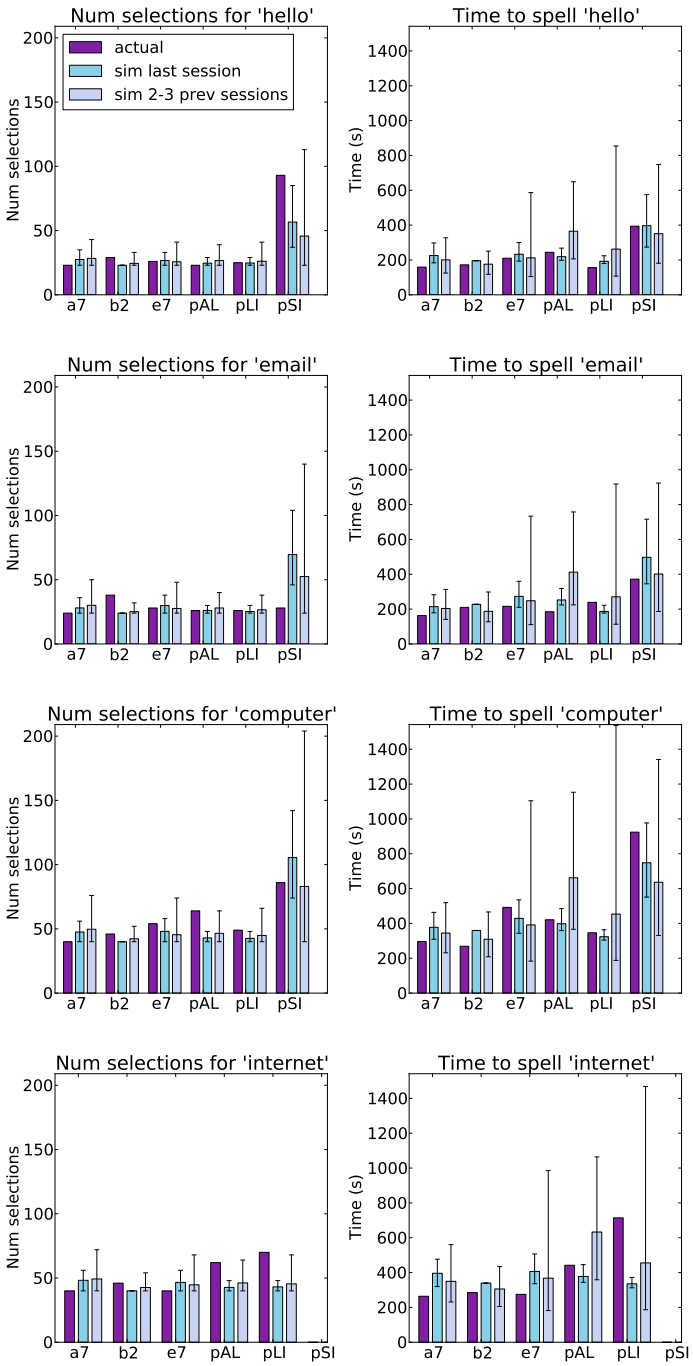


Figure 3.9: Actual and simulated prediction of number of selections (left column) and time taken (right column) to spell 5 words, for 6 participants (pXX denotes participants with motor disabilities). Actual (purple), average of simulated runs for calibration data from last session (blue, 95% PI) and simulated runs over calibrated data from last 2-3 sessions (green, 95% PI) are shown for each participant. Each participant spelled each word once. Note that pSI stopped after completing the word 'computer' and did not spell the word 'internet'.

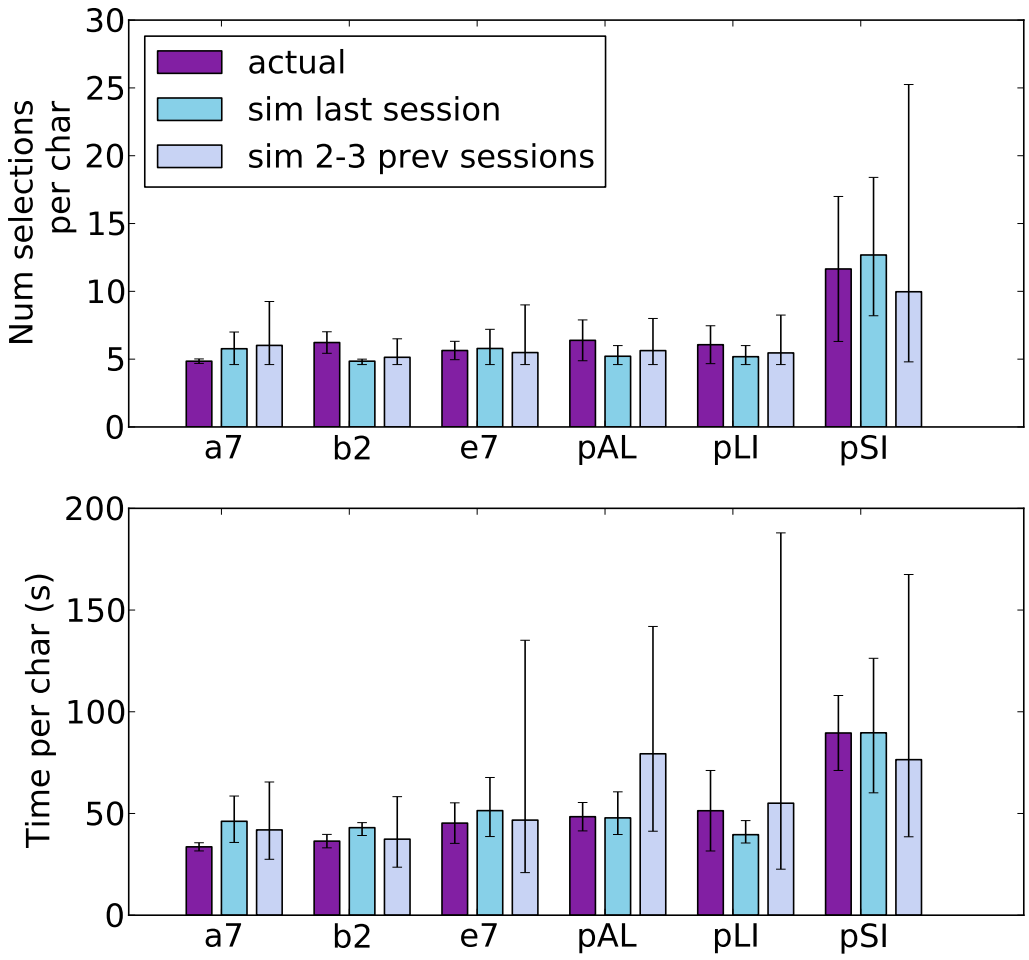


Figure 3.10: Actual and predicted values of the average number of selections (top) and time taken (bottom) to spell a character for 6 participants. Actual (purple, ±SD), average of simulated runs for calibration data from last session (blue, 95% PI) and simulated runs over calibrated data from last 2-3 sessions (green, 95% PI) are shown for each participant.

Table 3.4: Average time to select a character for actual and simulated trials (seconds). *Diff* is the *simulated* − *actual* times.

Participant	Actual (±SD)	Last session		2-3 Prev sessions	
		Mean (95% PI)	Diff	Mean (95% PI)	Diff
a7	33.6 (2.0)	46.2 (45.0, 47.4)	12.6	41.9 (40.5, 56.9)	8.3
b2	36.4 (3.3)	43.0 (42.8, 43.2)	6.6	37.4 (36.8, 42.0)	1.0
e7	45.3 (10.0)	51.4 (50.2, 52.8)	6.1	46.7 (45.9, 119.7)	1.5
pAL	48.4 (7.0)	47.8 (47.2, 48.6)	-0.6	79.3 (78.3, 84.8)	30.9
pLI	51.4 (19.8)	39.6 (39.0, 40.4)	-11.8	55.1 (54.2, 52.4)	3.7
pSI	89.6 (18.4)	89.7 (85.2, 95.4)	0.1	76.5 (71.3, 133.5)	-13.1
Mean	50.77	52.95	2.18 (MAE 6.29)	56.15	5.38 (MAE 9.75)

3.6.3 Discussion

A detailed analysis of the individual cases is interesting. For participant a7, the prediction consistently overestimated the number of iterations and the time to selection. This can be explained by the fact that in all 3 of the sessions prior to the experiment, the overall selection accuracy was less than 1.0, while during the spelling experiment, the participant's performance was perfect. This could be due to individual variation in performance between sessions. However, it is possible that some psychological factors arising from the nature of the task for this particular user led to an improvement in performance. For example, in the application, the time between selections (the time between selecting either left or right and the next time the feedback moved) was 3s, which is longer than is usually the case for the calibration trials. In addition, the participant chooses the direction of selection, instead of being told what to select. Thus, it is possible that the participant is better prepared to start the selection and thus performs better.

On the other hand, for participant b2 the prediction systematically underestimated performance in terms of the number of selections to select a word or character. Similarly, for participants pAL and pLI, the time taken to spell the words 'hello' and 'email' were within the confidence limits predicted from the last session, whereas for the words 'internet' and 'computer', the number of selections actually required by the user were in the upper limit of the prediction interval from the individual runs. This indicates that there was a higher error rate for longer words than there were for shorter words, which may be due to an increase in cognitive workload or stress.

The mean predicted times were the furthest off the mark for pAL, where the mean predictions from the individual runs over 2 sessions consistently overestimated the task times. It is possible that differences in the integration and threshold of selection may have been different between sessions, leading to an incorrect estimate in the time taken to make a

selection. For example, for participant pAL, the integration rate from previous sessions was 0.98, while in the actual experiment it was 0.95. This is a much slower rate of integration, with a minimum difference of around 2.5s per selection. Although a constant was removed from the total selection time, it is difficult to determine whether it sufficiently corrects the bias, as on one hand the figure is unlikely to be a constant, increasing as the calibration time increases, and on the other hand this may have improved the user's accuracy at the time. However, the difference may also have been due to variation in performance over different days, or to a difference due to the nature of the task, and thus it is useful to have a wider confidence interval which predicts the best and worst performances.

Finally, pSI did not manage to complete the experiment, choosing not to spell the word 'internet'. The time predicted for spelling the word 'computer' was 12.5min (actual 15.2min), which was far above the prediction for any of the other users for simulations with data calibrated from the last session (6.6min for pAL, although the less conservative estimate was 11.0min). On the other hand, the time predicted for pLI to spell the word 'internet' was underestimated by almost half the time (11.9min actual compared with 7.6min predicted from the last session), which may have occurred due to fatigue as this was the last word spelled. However, this was the only substantial underestimation of task times from the predictions. Thus, simulation can at least be useful for predicting when a user would be too tired to actually perform a task.

The number of selections required for a spelling task is generally used as a performance measure in optimising binary spelling tasks. However, as there may be a speed-accuracy trade off where the number of selections increases due to error, but do not proportionally increase in time (time per selection decreases), it is beneficial to provide an estimate of the likely task times for a participant. This may have been seen in participant pSI, as the number of actual selections did not generally correspond to the predicted task times. For example, in spelling the first word, the number of selections was much higher than predicted, but the task time was comfortably within the range of the predicted interval. This may have occurred for several possible reasons. Firstly, the differences could be due simply to individual variation in performance between sessions or misrepresentation of the base line performance. Secondly, the number of selections in the real data may contain a higher number of undos than in the simulated trials. This might happen as it was assumed that the user does not make mistakes with the undo. If mistakes were made while trying to select the undo, this would increase the number of selections. A related issue is a psychological one: the speed-accuracy trade-off may be due to a psychological effect of knowing that it is easier to correct mistakes, and the user might thus not try so hard for each selection, thus reducing the average time to select a character compared to the BCI calibration trials. On the other hand, the stress or cognitive workload may have led to an increase in the error rate, while motivation may have led in some cases to a decrease in error rate (as in with a7). Thus, additional data would be required to assess the contribution of various factors in the deviation from expected performance as predicted by the simulator.

The simulation analysis provides preliminary evidence that data from the session prior to an experimental session provides a useful ballpark estimate for a user's performance for a future session, in terms of the number of selections as well as the time required to achieve tasks. In general, the task performance during the calibration trials does not differ significantly from the experimental trials. Thus, simulation of task performance would be useful in providing estimates of the time it would take users to complete tasks, either for the purposes of experiments, actual application use, or communication with stakeholders. However, the 95% prediction interval which uses data from the individual runs over 2 or more sessions may better capture a best estimate of the best and worst expected performances. As the sample size is small, additional data would be required to provide additional confidence in the simulation results, to investigate further the reasons for deviations between simulation and actual results, and to determine which datasets and metrics provide the best prediction for the achievability of a task for a given user. It would also be interesting to compare the results with data that used a optimised speller.

3.7 General Discussion

In this chapter, it was demonstrated that offline simulations can be used to provide predictions of task performance in terms of the expected number of selections required and how long it would take for a user to carry out a task. Task performance between possible FSM structures, individuals and tasks can be compared and validated with real data. An advantage of using the simulation approach over analytic approaches is that it requires little or no mathematical knowledge, such that someone who is unfamiliar with graph theory or combinatorics can obtain predictions to improve or explore an existing menu design. It is also difficult to calculate expected performance metrics for specific tasks, but easy to generate these using simulations. Obtaining a distribution over the expected time it would take a user to complete specified tasks can help in planning experiments and ensuring that users are not too fatigued. Evaluating a task by the amount of time it would take a user to complete can also help the application designer by highlighting problems with the interface. If they know that it will take a long time to complete a task, the designer may well be motivated to streamline, simplify or customise the application further in order that a user can complete the tasks that are important to them in the shortest period of time and lowest effort possible.

The method used to design or optimise a menu hierarchy should depend on the application one wishes to develop. For a small number of letters in the alphabet and a uniform probability of selections, a variety of FSMs can be built and explored manually. A graphical user interface might allow a designer to drag and drop states and transitions to build and compare for task performance for a given set of user models. This can allow a designer to find 'good enough' solutions without requiring to compute complex calculations. As the number of letters in the alphabet increases, however, the number of possible configurations of FSMs increases rapidly, especially as non-uniform probabilities of letters and placement of back-up and delete nodes are taken into account. This makes it infeasible to explore

every FSM using the methods described. One approach to overcome this might be to break the large item selection into several sets of smaller menus, and select the combination of menus that lead to the best performance. Simulation is also used in combination with other techniques. For example, theoretical predictions are usually validated with with simulations (e.g. Bianchi et al. (2007)), and it would be possible to select the best tree structures out of a set that has been narrowed down using optimisation techniques. Here, a further benefit of using simulations is that the variance in task performance can be easily obtained, which cannot be so easily derived.

The current chapter explored only a single user model, assuming that there are no cognitive effects of the user interface on the user's performance. More complicated models can be used which update the selection accuracy and time to selection, such as where a user's performance is likely to deteriorate after some period of use, or frustration with the system after being stuck in a menu loop for a period of time. The user's level of motivation can similarly be built into the model by increasing or decreasing the selection accuracy and time-to-selection accordingly. It is possible that this would improve the accuracy of the predictions with respect to real data, and enable a designer to further explore the expected benefits and costs of particular menu hierarchies.

3.8 Conclusions

In summary, this chapter highlighted the benefits of using offline simulations of a discrete, binary control paradigm to investigate user performance in different control settings. A benefit of simulation rather than analytic approaches to estimating a theoretical expected value is that simulation can provide estimates of the expected distribution and range of task performance, in particular for how long it will take to complete a task. In addition, the expected variance of task performance is captured easily, which adds information to the expected performance. It was shown that the speed of individual selection times as well as the accuracy can affect the bit rate, such that the overall task time can be shortened if the TTS can be reduced to one or two seconds at a lower selection accuracy. Comparison of the offline predictions using simulation with online BCI performance shows that the results are comparable, validating the use of offline simulations for the estimation of task times and task performance. Providing that there are no cognitive issues or difficulties with navigating the user interface, the offline estimations are a good indicator of task performance.

4 Simulating the feel of Continuous Control

Summary. This chapter describes the development and validation of a simulator which models the classifier output of a specific 2-class Motor Imagery system, discussing the benefits and limitations of different signal generation and simulation techniques that were explored. Whichever method is used, the goal is to flexibly reproduce signals that model a particular user, in order to obtain some useful information about a system. In terms of research, modelling the user may lead to new insights about the nature of Brain-Computer interaction, while simulation can also be useful for designing applications or optimizing parameters for a control paradigm to achieve optimal performance.

4.1 Introduction

Chapter 3 investigated the value of simulating discrete selections to predict performance, as measured by the selection accuracy and time taken to make a decision. Although useful, this level of simulation is limited in that it is useful for offline analysis but does not allow one to explore other types of control paradigms, or to understand what it might actually feel like to use a BCI system. This chapter describes the methods used to build a simulator whose purpose is to capture the properties of BCI control that are important to design and to the subjective feel of using the interface. The argument is that such a simulator can be used to reduce the cost of developing and designing applications for BCI. As elaborated upon in Chapter 2, this can be done either by carrying out offline simulations to predict performance, or by being used in an online simulation mode with a human being in the loop. The online simulation mode is useful for

1. providing designers with a tool that enables them to experience what it is like to use a BCI, without needing to actually use a BCI

2. providing a means for communication with stakeholders, enabling them to understand how a BCI works and feels

3. allowing developers to develop and debug BCI applications easily, as real BCI input can be directly replaced by the simulator

4. performing usability tests of a novel control paradigm or application without users requiring to use a BCI.

The sections of the chapter follow through the process used to build the simulator, detailing what is being simulated, the model assumptions, signal measures that have considered, modelling approaches used and the evaluation of the resulting models.

4.2 The level of simulation

In considering the level at which the output of the simulator should be aimed, its purpose should be taken into careful consideration. While simulation at a high level may be less costly, important details may be missed. Conversely, simulation at a lower level may be more complex or difficult, incurring unnecessary costs if it is more complex than required for the purpose. In an MI-BCI, the following levels, arranged from high to low, can be distinguished as follows:

- **EEG**. Simulation at the level of EEG has been investigated to improve signal processing, artifact removal, feature extraction and classification techniques in BCI (Tangermann, 2012). Building a model of the EEG has also been used to understand the origin of neurophysiological signals, such as the ERP (Yeung et al., 2004). Yet it may be argued that simulation at this level is overkill for the purpose of designing and developing applications for BCI, as the complexity of EEG and number of signals (one signal for each electrode used) needing to be simulated, may reasonably incur greater time costs than simulating a signal that incorporates this information. Modelling the control characteristics of interest becomes more complicated when having to consider these aspects as well as the signal processing and feature extraction tools required to obtain a useful signal. For our purposes as a tool for design and development, simulation at higher levels may thus be easier and more valuable, if the important information can be captured.

- **Classifier Output**. The signals resulting from 'cleaning up' the EEG are typically fed into a machine learning classifier whose output is a single value or vector of values. These can be used by an application or control paradigm and are typically generated at a constant rate of several per second. As the output of the classifier is noisy (Figure 4.1), it is typically integrated (accumulated) in some way over a few seconds; this allows time for the system to gather evidence about the user's actual intent. Simulation at this level is interesting as it can potentially be used to explore or optimize parameters for different integration methods. Relatedly, it can also be used to explore new control paradigms that would rely on the classifier output at this level. A complexity is that the type of classifier used must be taken into account in order to provide input that a BCI application expects. Figure 4.1 provides examples of the output of 3 different types of classifiers which give different ranges of signals. A probabilistic classifier will have real-valued outputs between 0 and 1, while an LDA will have a output as a measure as a distance from some hyperplane which separates the two classes. An output simply labelling the discrete classes may also be used. The distinction is important as it constrains the measurements and methods that can be

used.

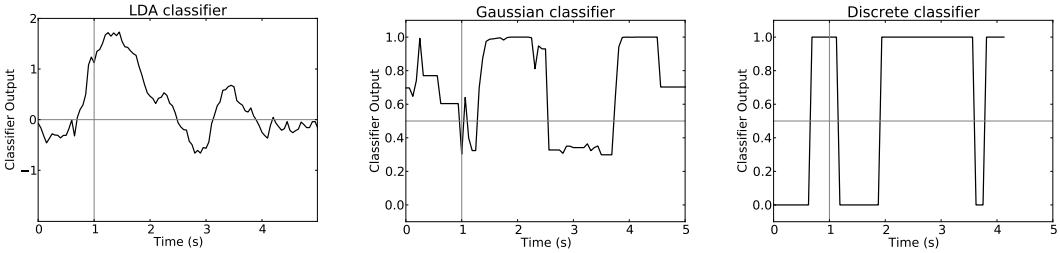


Figure 4.1: Examples of classifier output from different classifiers, showing the difference in output values. At $t = 0$ the target is shown to the user, while at $t = 1$ the feedback cursor begins to move. Left: LDA classifier; output is a distance from 0. Centre: Probabilistic classifier as used in this thesis (Millán et al., 2004); output between 0 and 1. Right: Discrete class decisions; output is 0 or 1 at each time step. Note that the three graphs are data from real experiments, taken from unrelated trials.

- **Integrated Classifier Output**. As previously mentioned, the classifier output is accumulated over time by some method of integration. This enables the system to gather evidence about the user's intent before making a discrete decision. This is the value that is usually mapped to whatever feedback is presented to the user given a control paradigm, and is therefore directly related to the feel of BCI control. Again, to directly simulate the output at this level, one must take into account the type of integrator that is used. Figure 4.2 shows the different characteristics of two integration methods. In the subfigure on the left, the speed of the feedback will be variable, while the speed of the feedback produced by the stepped integration method on the right will be constant. Properties of the feedback such as the speed of movement may lead to a difference in the perceived or actual control the user has on the system; this is as yet an unexplored area of research in the BCI literature.

- **Discrete decisions**. Thresholding the value of the integrated classifier output leads to the system's decision about the selected class. Again, simulation at this level was the focus of Chapter 3, and as previously mentioned, does not capture level of detail required for simulating the feel of continuous control. However, the output of the classifier certainly gives rise to this and as such it will form one of the measurements of the quality of the simulator.

Thus, the outputs chosen for the simulator are 1) the output of the classifier and 2) the integrated classifier output. These are most relevant for the purpose of simulating the feel of control of a BCI, as these stages result in the accumulated effect of the processing at the lower stages. Furthermore, as the integrated classifier output is usually straightforward to calculate, simulation at this level can easily be used as input to an application. The remaining factors in deciding which of the two signals to directly simulate pertain either to which method provides a more realistic or useful output for the topic of interest, and which are simply easier to use.

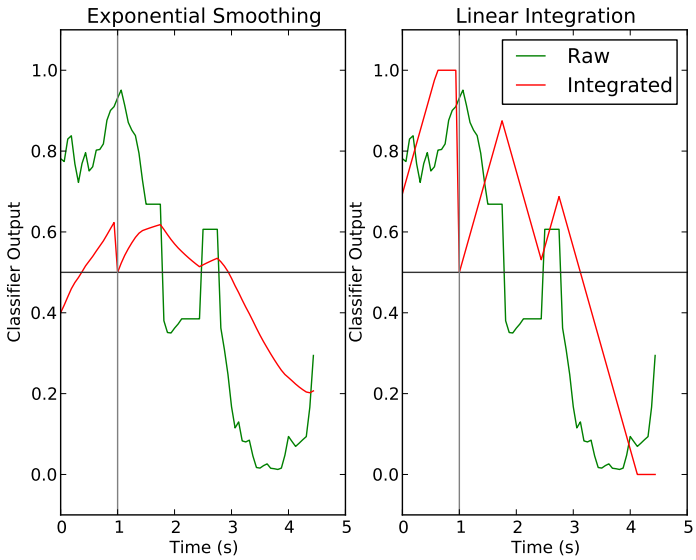


Figure 4.2: Integration of classifier output.

Another way of looking at the level of simulation is the input method used to simulate the BCI system. As there is other input method that shares the same characteristics (hence we try to simulate it!), suitable inputs must be selected. Keyboard input was chosen in the current work as it provides a simple discrete input to the system. Pressing the shift keys correspond to either imagining movement of the right hand or left hand. Although this requires the user to use both hands, it does not provide the user with the sense of continuous control, nor does it generate any uncertainty (from the user's point of view) about what the input to the system is. Again, this demonstrates the cost-benefit-stage-of-simulation trade off that is required in making modelling decisions.

4.3 General Model Assumptions

Understanding of the BCI process and levels of simulation, and the development of the following model assumptions, were made possible not only by becoming familiar with current literature, but also by consulting with BCI researchers. In particular, the conceptual model of the process (Figure 4.3) was built by sitting down with a BCI expert and clarifying the stages in the control loop of a BCI. At each stage of this model, the control characteristics arising from that stage were identified, which were then used to develop the resulting assumptions. For this 2 class MI system, 3 states left MI, right MI and idle are defined corresponding to the mental state the user intends to be in (Figure 4.4). The idle state covers all other mental states where the user is not intentionally trying to control the BCI. Delay in the system arises due to several sources at different stages. These can be modelled either as a constant, or variable delay in switching between mental states.

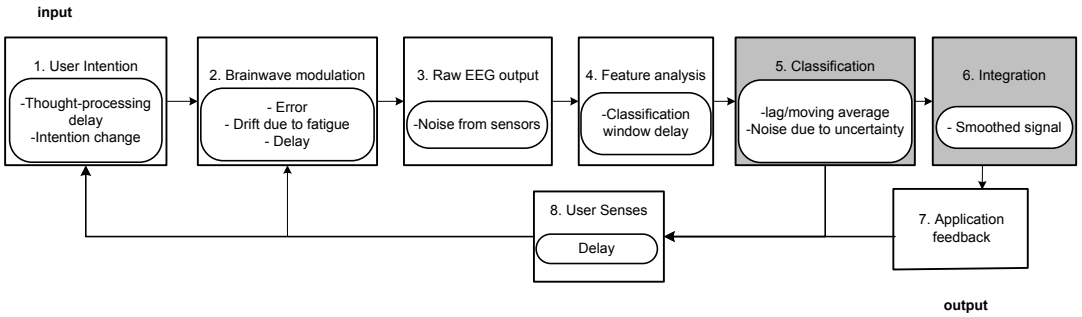


Figure 4.3: Stages in the process of an MI-BCI system, displaying the control properties arising from each stage. Grey boxes denote the stages at which the output is simulated in the simulator described in this chapter.

Not everything in the conceptual model is accounted for with the current simulators. In this first approximation, loss of control that would occur for some people after trying to imagine the same movement for an extended period of time is not considered. The closed-loop feedback of interaction between the subject's thoughts and the influence of feedback, as well as the longer term deterioration in performance that would occur due to fatigue or stress (Blankertz et al., 2010), are also not modelled. However, the flexibility of the simulator allows for additional simulation blocks or tools to be added as the model is improved. Additional assumptions must be made specific to the methods and approaches used and are stated as they arise in later sections.

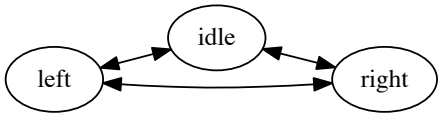


Figure 4.4: States identified for a 2-class MI-BCI, representing the intentions of the user. In the 'idle' state, the user does not wish to select anything. The left and right states correspond to two mental classes which is usually a combination of left hand, right hand or feet motor-imagery. In this representation, the user can switch between the 'idle' state and a mental class, or directly between mental classes.

4.4 Measures of BCI Signal Properties

In order to describe, analyse and compare simulated and real data, quantitative measures must be employed at different levels of the system. Some of these measures may be used to evaluate or validate the models, while others will be used to develop the models. Graphical plots are used to visually analyse the data for patterns and possible models, while statistical

tests aid in determining the fit of the data to models and distributions.

The selection accuracy (defined here as the probability of the integrated classifier output to terminate at a certain threshold) and time-to-selection have been described in chapter 3. In addition to these, metrics used to describe the trajectory of the [integrated] classifier output should be used to capture the properties of the time series, although it is as yet unclear what quantitative measures map to the user's perception of the feel of control. Standard techniques for time-series analysis, which can be categorised into *time domain* and *frequency domain* analysis. Examples of time domain analysis are the autoregressive moving average model (ARMA) and autocorrelation, while an example of frequency domain analysis is the Discrete Fourier Transform.

The *Fast-Fourier Transform* (FFT) is an algorithm used to represents a signal in the frequency domain (a very clear and practical textbook on the subject is Smith (2002)). Essentially, any signal can be broken down into, and reconstructed from, a sum of sine waves whose frequencies are between 0 and half of the sampling rate of the signal. The real FFT can be represented in two ways. The rectangular notation represents the signal as a set of scaled sine and cosine waves, while the polar notation stores the magnitude (amplitude) and phase (angle) of the set of cosine waves. The frequency components of the signal relate to the feel of control of cursor feedback as this influences the oscillations in the feedback; for example high frequency components would produce fast back-and-forth movements in the cursor, while low frequency components can represent the drift of the signal.

Autoregression models a stochastic signal by assuming that that each signal at time t is a weighted sum of previous observations plus noise (and optionally a constant), such that the signal at X_t is given by

$$X_t = \sum_{i=1}^{p} \theta_i X_{t-i} + u_t$$

where p is the order of the model, θ_i are weights and u is a random variable drawn from an independent, identical distribution. The moving average model

$$X_t = \sum_{i=0}^{p} \theta_i u_{t-i}$$

assumes that the signal at X_t is a combination of the previous inputs. The Box-Jenkins methodology is a widely used methodology for estimating the parameters of the model for a particular signal.

Autocorrelation is a measure of how samples in a signal are related to one another. If A is the signal and A_t is a sample at time t, then the autocorrelation of the signal with lag k is the degree to which samples A_t vary with A_{t+k}. The autocorrelation gives a measure of randomness in the signal. The classifier output should covary at least at $k = 1$ since the output is essentially a transformation of a moving average of the raw EEG signals.

Distributions of the *position*, *velocity* and *acceleration* of the time series may be used to compare the classifier output signals. The classifier output is directly related to the selection accuracy of the user and may influence the speed and acceleration of the feedback if an exponential smoothing integrator is used. The speed and acceleration of the integrated classifier output are directly related to the feel of control of the system. One can speculate that a constant speed and acceleration may enable the person to feel more in control, and if speed and/or acceleration is too fast, the user may feel that they are not in control. These can represented as histograms where the actual values are binned into ranges of values. The histograms may be compared using the Kullback–Leibler divergence (Kullback and Leibler, 1951) which is an information-theoretic measure of the difference between distributions given by

$$D(P||Q) = \sum_i P_i \log \left(\frac{P_i}{Q_i} \right)$$

where P represents the real data is the and Q the simulated data. All of these methods make the assumption that the signal is stationary; that is, the properties of the signal do not change over time. For this first exploration into simulation of the classifier output of a BCI, this chapter makes the assumption that this is the case within each mental class or state. In reality for a BCI signal, this is unlikely to be the case for several reasons. Firstly, the feedback likely influences the classifier output as a function of how much control the person has and this is not taken into account. Secondly, the properties of the signal may change in some time-dependent manner that is more complex than is currently represented.

4.5 Modelling Approaches

Simulation modelling approaches can be categorized into those that are *generative* or *data-driven*. A generative approach begins by building a mathematical or other computer model of the system. Parameters of the model can be fit to data either manually or by an optimization algorithm. The benefit of this approach is that by changing the model parameters, one can easily generate simulated data for individual characteristics that are possible, but for which no data are currently available. The approach also allows for testing model assumptions and hypotheses to understand a specific phenomenon. By contrast, a data-driven approach simulates a system by producing a large number of transformations of real data. A benefit of this might be that it is less complicated in terms of fitting the model parameters. It should be noted that a complex simulation model may have multiple components that use both approaches.

Another way of looking at approaches to modelling are *top-down* and *bottom-up* approaches. These define the starting point of the model: a top-down approach begins with modelling the high level features followed by the low level features, while a bottom-up approach looks at the low level features which produce the high level characteristics. In our specific case, a top-down approach would model the system at the level of discrete selections (accuracy and time-to-selection), followed by the time series of the integrated and pre-integrated signals.

Two examples of bottom-up techniques are now described. These attempt to capture the low-level feel of the system by simulating the (pre-integrated) classifier output over time. An example of a top-down, data-driven approach might be to add noise to the average time series of the integrated classifier output over some number of trials. A top-down, generative approach might instead start with a model of the linear trend taking into account the high level summary statistics of performance.

4.5.1 A bottom-up, generative approach using a Markov Chain model

A Markov chain is a process where at any discrete point of time, it is in one of the states $\{s_1, s_2, s_3...\}$ in the set S. At each time step, there is a probability of either staying in the same state or transitioning to another state in the set, such that for each state s_i, there is a probability p_{ij} of transitioning from s_i to s_j in the next time step. In the current model, a Markov chain is defined for every *intentional state* in the 2-class MI-BCI, left, right and idle. The assumption in each state is that, at a given time step, the classifier believes that the features belong more strongly to one class (*left*) or the other, *right* (i.e. the output of the classifier ranges from 0 to 1, but leans towards either 0 or 1). Thus, in a given intentional state, there are two states in the Markov Chain, s_1 and s_2, and two transition probabilities p_{12} and p_{21}. The values of the classifier output in each state in the chain are drawn from distributions D_1 and D_2, which can be any function that generates random variables whose values could plausibly be generated by the classifier. Note that since a 2-class MI-BI does not detect a third, idle state, any difference in the behaviour of the states must be defined in the same way as the intentional states, either by changing in the distributions of the Markov chain states D_1 and D_2 or the transition probabilities p_{12} and p_{21}. Figure 4.5 illustrates the Markov chain for one of the states.

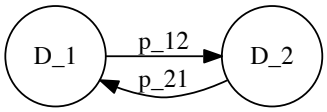


Figure 4.5: Markov chain representing the parameters for one intentional state (left, right, or idle). At the current time step, a value is generated from the distribution D_i from the current Markov chain state s_i, and the probability of transitioning to the next state at the next time step is given by p_{ij}.

In the current model, a delay is applied, which accounts for the pure time delay due to the feature extraction and classification process, which was estimated to be 0.5 seconds (since every classifier output took into account the previous 1 second, it seems reasonable that a change in state would begin to be seen from around half this time). A further delay occurs when switching from one mental state to another. This is difficult to measure, as it is difficult both to pinpoint exactly when the user began to switch mental states, as well as

when exactly the peak intensity of the other mental state was reached.

The parameters of the model can be selected either by manual tuning, by using an optimization algorithm, by calculating the parameters or a combination of the three methods. Section 4.6.1 details this methods for choosing parameters of the model that were used in the current analysis.

4.5.2 A bottom-up, data-driven approach using the IAAFT algorithm

One method of simulation is to generate surrogates of the signals. These are signals that share some of the same statistical properties with the original signal but differ in the actual time series. For our purposes, this enables us to generate signals that have similar properties to the original signals but are not exactly the same. The hypothesis is that this will capture the properties of the feel of the signal, as well as the selection accuracy.

The Iterative Amplitude Adjusted Fourier Transform method (Venema et al., 2006) was used to generate surrogates of the classifier output for each trial. The algorithm produces surrogates of a time series by preserving the values of the original signal (template) and the magnitude of the Fourier spectrum, while shuffling the phase of the Fourier spectrum of the real data. This has been used, for example, to simulate cloud distributions for weather predictions (Venema et al., 2006). As the matlab code for the algorithm is laid out in the paper, the following represents a qualitative step through of the algorithm:

```
1. create a surrogate of the same length of the template from white noise

2. continue until max iterations reached OR convergence criterion reached:

        a. replace the FFT magnitude of the surrogate
        with the FFT magnitude of the template
                (preserve the phase of the surrogate)

        b. replace the values of the surrogate
        according to the values of the template
                in order of rank
                (this again changes the FFT power spectrum)

        c. calculate the convergence criterion as a
                change in the accuracy of the FFT magnitude of
                the surrogate compared with the magnitude of
                the template

3. return surrogate
```

In this way, one ends up with a surrogate that is the same length as the original signal, whose values are the same as that as the original signal but shuffled in such a way that the overall frequency content is very close to the original signal. For trials where the surrogate signal has not reached the selection threshold by the end of the last sample, a new surrogate is generated (either from the same signal or a different one) and the simulation continues.

4.6 Evaluation

Validation of a simulator is required to provide confidence in the model such that it can be used to answer unknown questions of interest. Since the simulation model will always be an approximation to the real system, one perspective on validation is that the goal should be to establish confidence in the applicability of the model for its intended objectives (Sargent, 2010). The level of confidence in a model increases as the number of tests are passed; however, at a certain point the degree of this increase slows as the cost of the testing also increases. Thus, in the ideal case the modeller should select a battery of tests that provide the highest degree of confidence with the lowest overall cost. Once the user is satisfied that the essential parts of the model have been validated, the simulation model can be used. In the current case, the model's outputs can be thought of as being in two domains requiring separate analysis techniques; offline evaluation compares the real data with the simulator's output in terms of summary statistics and graphical plots, while online evaluation involves users comparing the subjective feel of the feedback control in an online paradigm. In each domain, levels of validation with increasing sophistication can be identified, from face value judgments (comparison by looking at the output) to statistical tests.

4.6.1 Offline evaluation

Offline evaluation involves comparing the output of the simulator with real data, either quantitatively using numerical and statistical methods, or visually using graphs and diagrams. Quantitative measures were used to compare the real data and data generated by the different the different simulation methods. In particular, the accuracy, time-to-selection and time series of [pre-]integrated classifier output were compared.

Models

Three methods of calibrating the parameters for the Markov chain model, and the IAAFT method, were used to generate simulated data for binary trials for 9 participants. For each participant, 30 trials (2 consecutive runs) were used and simulated data was generated for both left and right trials separately. As previously described, the signal for each trial was assumed to be static and the samples from 1 second after the cue indicating the target were used to generate classifier outputs at 16Hz according to the BCI system described in Chapter 3.2. The linear integration method was selected by the author at the time of data collection. Following this, at time $t = 0$, the target was 'shown' to the simulator, setting the intentional state of the simulator to the target. The step integrator was used, which

set a minimum time-to-selection at 1 second from the time the feedback started. Thus, at $t = 0$ the integrated output was reset to 0.5, and at each sample from $t = 1s$ (sample 16), the integrated classifier output was incremented by 0.03125 if the classifier output was $\geq$ 0.6, and decremented by the same step size if the classifier output was ≤ 0.4. The end of a trial was reached when the integrated output reached either 0.0 (left) or 1.0 (right). As the means for the accuracies were very close at 100 runs (30 trials each), this number of runs was used to generate data for each simulation model.

The first method of selecting the parameters for the Markov chain model (denoted *Markov Pmf*) drew classifier output values from the probability mass function (PMF) of the real data for the states in the model. The probability of switching between states was given by the conditional probability

$$P(S_{t+1} = l|I, S_t = r) = \frac{N_{IS_{t+1=l},S_{t=r}}}{N_{IS_{t=r}}}$$

and correspondingly for $P(S_{t+1} = r|I, S_t = l)$ where I is the intended class, l and r are the left and right states of the Markov model, S_t is classifier's 'perceived' state of the user at time step t and S_{t+1} is the perceived state at time step $t + 1$. Classifier output values ≤ 0.5 are taken as state l, while values ≥ 0.5 are taken as state r. $N_{S_{t+1=l},S_{t=r}}$ is the number of trials where the next trial is in state l, and $N_{S_{t=r}}$ is the total number of trials in state r. The second method (*Markov Prob*) is almost the same, apart from using a Beta distribution to model the values for each state of the Markov model. This has the advantage that the values of the model are captured with a small number of parameters. Again, the assumption is that the distribution of the output at sample $t + 1$ is independent of the output at t. Although this is not true in reality, this simplifying assumption was made as it seemed a reasonable assumption for a first-pass simulation.

The first two Markov chain models and the IAAFT do not take into account that there may be a variable delay in the time taken to switch between mental states. Thus for comparison, a third model (*Markov Delay*) was used where a delay modelled as a Gamma distribution was applied at each intentional state change. The parameters for the switching probabilities, $P(S_{t+1} = l|I, S_t)$ and the delay distribution were optimised using the Differential Evolution algorithm (Storn and Price, 1997; Das, 2011), a global optimisation algorithm. The cost function used to compare the model output during optimisation was the sum of the difference in accuracy between the real and simulated data, the KL-divergence of the acceleration histograms of the integrated classifier outputs, and the KL-divergence of the time-to-selection histograms.

Results

Selection accuracy. The selection accuracy is the percentage of trials where the integrated classifier output reaches the correct target threshold. As can be seen in Figure 4.6 which compares the selection accuracies for real and simulated data for each left and right target for each participant – *data points* – the trend of the real data is followed rather closely

by the simulated data. Box plots summarising the difference between the mean selection accuracy over the simulated runs and the actual accuracy, for each model, are shown in Figure 4.7. Table 4.1 shows the mean error (ME), mean absolute error (MAE), error range and the percentage of participant scores that fall within the 95% prediction interval (PI, i.e. the within the 2.5th and 97.5th percentile range of the data). The real accuracy falls within the prediction interval for all of the data points, while the MAE is remarkably close to the actual data at around 2-3 percent deviation from the mean for each of the models, with the highest error being 11% with the Markov Delay model.

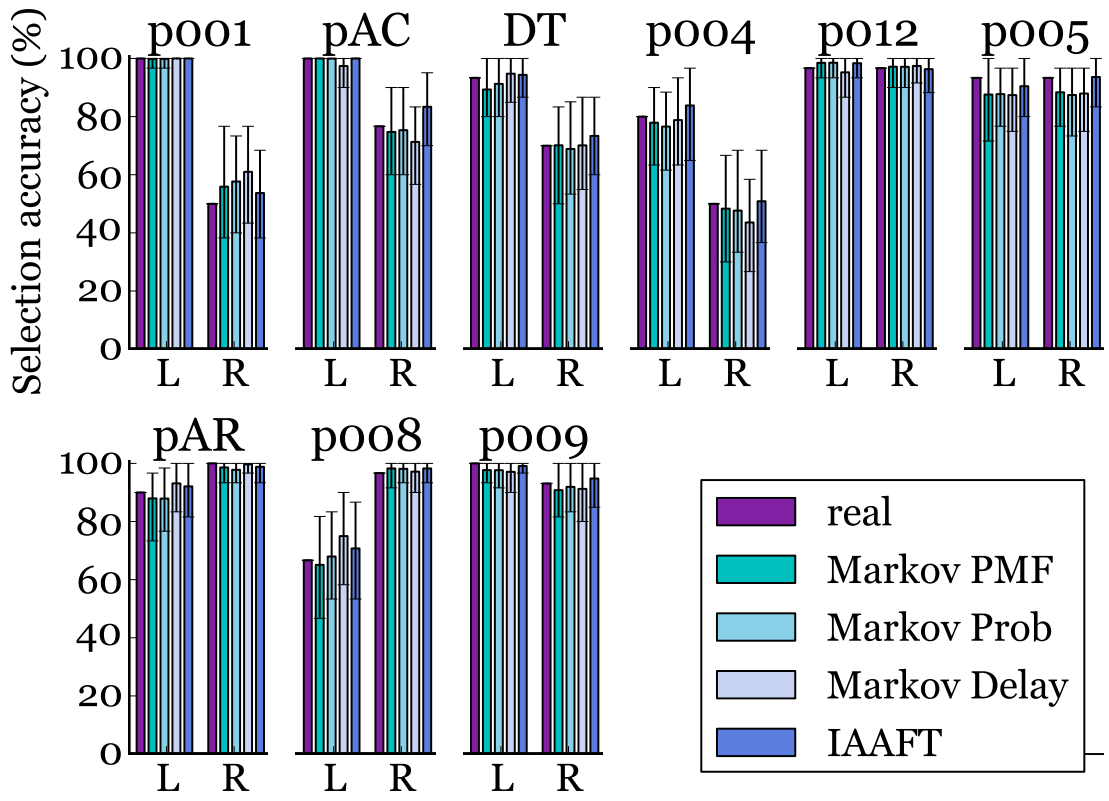


Figure 4.6: Selection accuracy of actual and simulated data (mean over 100 runs) for different participants using 1) the Markov chain method using the probability mass function (PMF) of the real classifier outputs (*Markov PMF*), 2) the Markov chain method using the Beta distribution of the real classifier outputs (*Markov Prob*), 3) the Markov chain method optimising the probability and delay in switching states (*Markov Delay*), and 4) the IAAFT method (*IAAFT*). Error bars are the 95% PI over the runs.

Time to selection. Figure 4.8 shows examples of time-to-selection histograms for the real data as compared with simulated, using bin widths of 0.5s. Overall, again the simulated timings match well with the real data as for the most part, the real data is contained in the percentile intervals of the model. Interestingly, for many data points there appears to be two

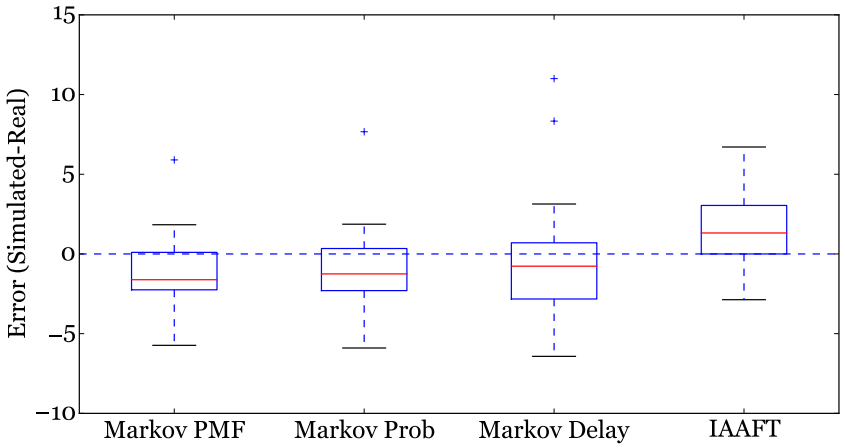


Figure 4.7: Box plots of the difference between the mean simulated accuracy (over 100 runs) and the actual accuracy for the 4 simulation models. Each box plot represents 18 data points (9 participants × 2 MI classes.)

Table 4.1: Comparison of accuracies for simulated and real data for 4 simulation methods, showing the mean percent error (ME), mean absolute error (MAE), minimum and maximum deviation from the mean (Error range), and the percentage of scores contained in the prediction interval (PI).

Simulation Method	ME	MAE	Range	Scores contained in PI
Markov PMF	-1.12	2.23	(-5.7, 5.9)	100.00 (18/18)
Markov Prob	-0.94	2.36	(-5.9, 7.7)	100.00 (18/18)
Markov Delay	-0.46	3.26	(-6.4, 11.0)	100.00 (18/18)
IAAFT	1.43	2.02	(-2.9, 6.7)	100.00 (18/18)

or three peaks in the time histograms of the real data, indicating a multimodal distribution. This could be further investigated by analysing a larger number of trials. Interestingly, some of these peaks are somewhat captured by the trends in the upper percentiles of the histograms, especially in the IAAFT model, but never completely in the averaged simulation runs.

Table 4.2 shows the ME, MAE, mean error range, median error (MedE), median absolute error (MedAE) and the median error range, while Figure 4.9 and Figure 4.10 show the comparison of mean, median and spread for each data point. Figure 4.9 shows the *average* range (top) and 95th percentile range (bottom) over the simulation runs (i.e. average for 30 trials), while Figure 4.10 shows the range and percentiles taking all the simulation trials as a single set. It can be seen that even though the overall maximum time to selection of the simulated trial is longer than the real data, the percentiles fall short of the real data. Thus, on average, the Markov models underestimate the means and medians, reflecting that the average range is shorter than the real data. That the medians are closer to the real data than the means for the Markov models (e.g. MedAE = 0.31s, MAE = 0.52s for the Markov Prob model) is accounted for by the fact that the peaks in the distribution are not captured well in the simulated data. While the IAAFT model does a better job at matching the means and medians, it can be seen that in most cases (e.g. p004 left, Figure 4.8, top right) the closeness to the mean is due to a longer tail in the time-to-selection histograms than in the real data. Finally, the optimization model Markov Delay sometimes improves the match in the time-to-selection as compared with the other Markov models, but inconsistently, thus the overall effect is that the MAE and MedAE are similar to the other Markov models, but the variance is less skewed towards underestimation.

Classifier output time series. To compare the trajectories of the pre-integrated and integrated classifier outputs, trials of similar length as the first and second peaks of the real trials for 2 participants (*right* target) are shown in Figure 4.11. On face value, it appears that the Markov Pmf and Markov Prob models match the integrated classifier output fairly well in terms of the number of turning points or oscillations, while the output from the

Table 4.2: Comparison of timings for simulated and real data for 4 simulation models, showing the mean error (ME), mean absolute error (MAE), minimum and maximum deviation of simulated means from the real mean (Error range), and similarly the median error (MedE), median absolute error (MedAE), minimum and maximum deviation of the simulated medians from the real median.

Simulation Method	ME	MAE	ME range	MedE	MedAE	MedE range
Markov PMF	-0.52	0.52	(-1.09, -0.20)	-0.31	0.32	(-0.73, 0.12)
Markov Prob	-0.52	0.52	(-1.06, -0.20)	-0.31	0.31	(-0.73, -0.03)
Markov Delay	-0.46	0.51	(-1.11, 0.46)	-0.24	0.31	(-0.73, 0.61)
IAAFT	-0.07	0.22	(-0.64, 0.41)	0.11	0.22	(-0.21, 0.73)

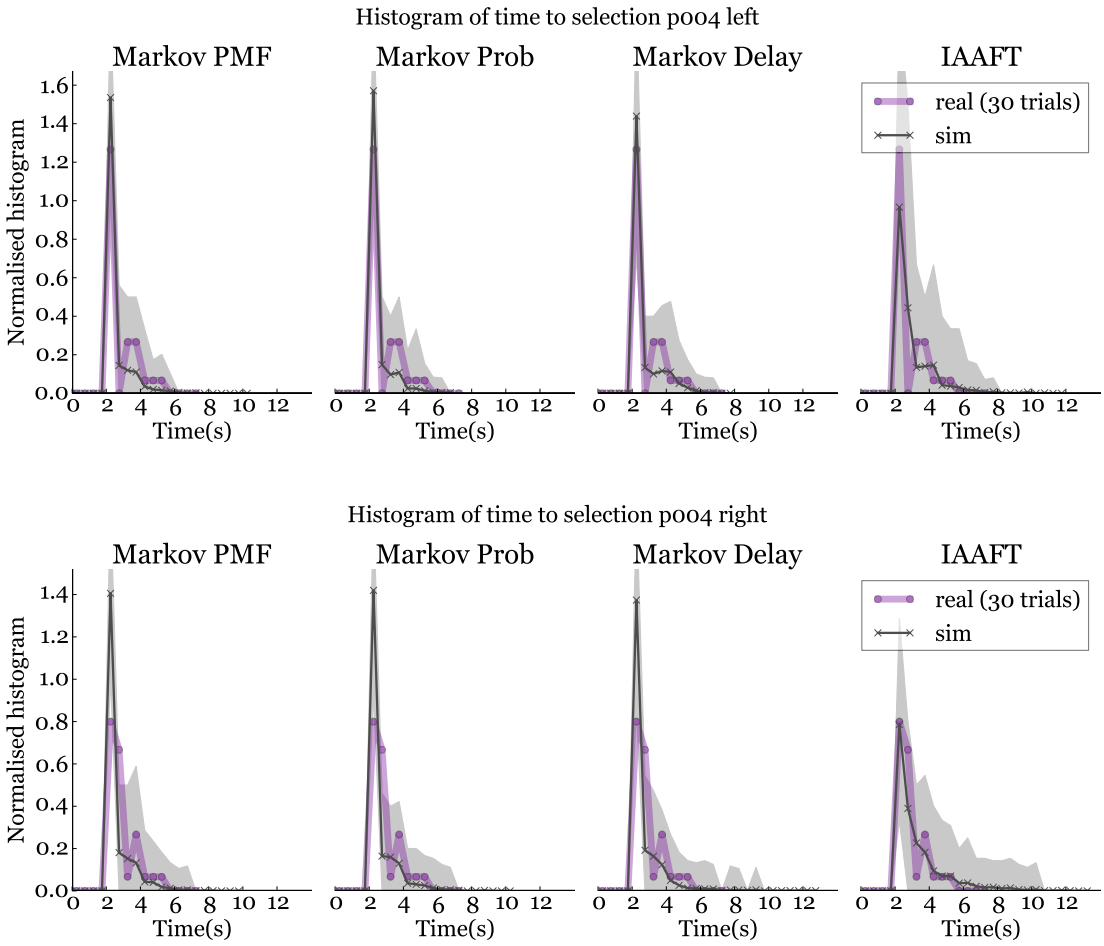


Figure 4.8: Histograms of time-to-selection for real (purple) and simulated (grey, 95% percentiles averaged over histograms of 100 runs of 30 trials each) data over different simulation models, for one representative participant. The histograms have bin widths of 0.5s. The histograms show that the main peak is the same in both real and simulated data, tailing off as expected for task times. Subsequent peaks in the real data are sometimes partially captured by the average simulated data, most often for the IAAFT model.

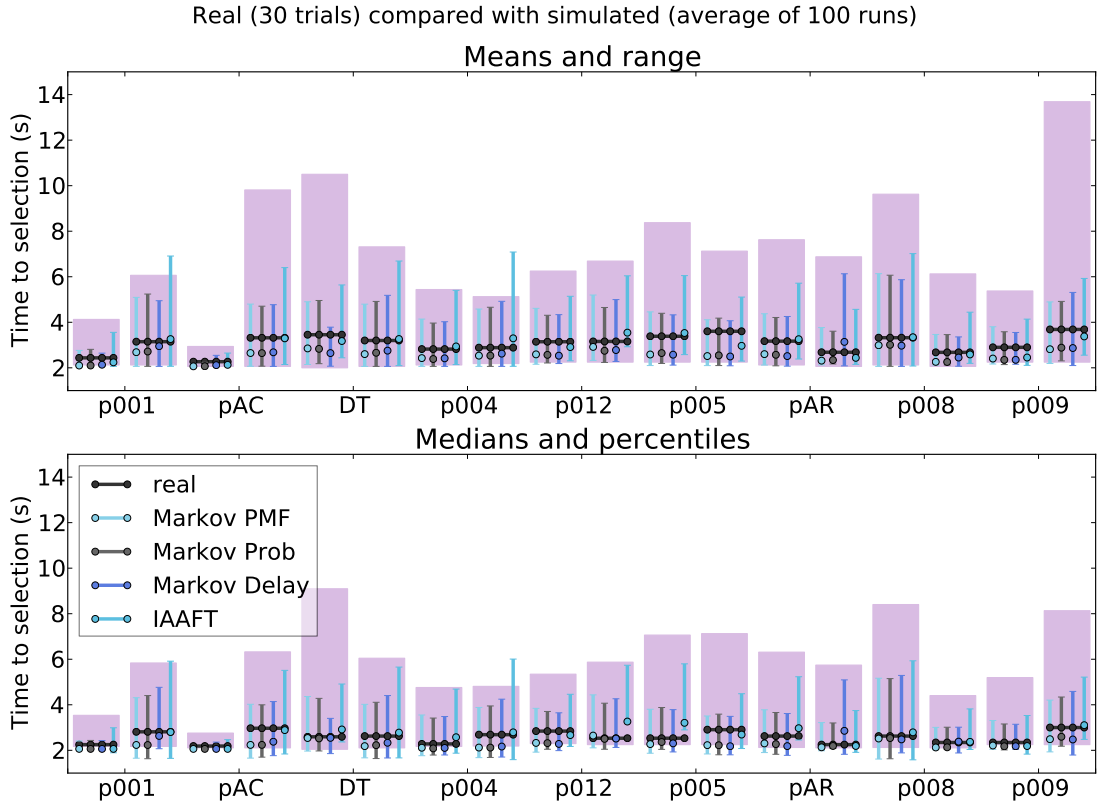


Figure 4.9: Comparison of timings for real (purple fill, black dots) and simulated means and range (top) and medians and percentiles (bottom). The real data for each data point consists of 30 trials (2 runs of 15 trials each). The simulated ranges and percentiles are the average over 100 runs of 30 trials per run.

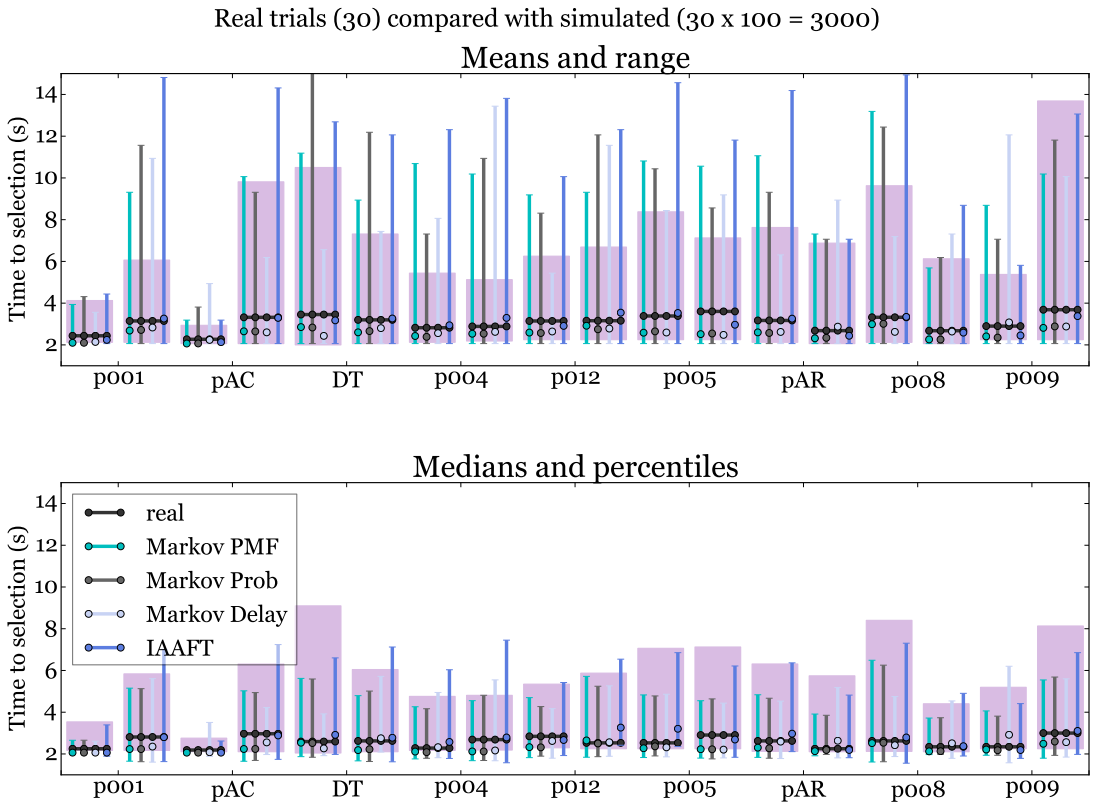


Figure 4.10: Comparison of timings for real (purple fill, black dots) and simulated means and range (top) and medians and percentiles (bottom). The simulated ranges and percentiles are the values from the total of $100 \times 30 = 3000$ trials.

IAAFT model produces integrated classifier outputs that are noisier than the real data and Markov models. The number of turning points and the time between turning points could be used to quantify the level of noise for the step integrator, while the FFT could be used to characterise the noise for an exponential smoothing integrator.

Discussion

From the offline comparison of model data with real, it has been shown that simulated offline data based on the classifier output can produce signals that match the real data to a large extent in terms of selection accuracy, time-to-selection and classifier output time series. The different models are similar with slight differences in that subjectively, the time series of the step integrator look more like the real data, while the IAAFT model is better able to match the time-to-selection (and slightly better match the selection accuracy). The optimization algorithm is sometimes able to improve the fit of the Markov models to the data, but the current implementation is inconsistent. Thus, it may be that the Markov models can be tuned to the subjective control in an online experiment, while the IAAFT model as it currently stands can be used in offline analysis of data. In terms of offline analysis, it would be useful to find out if the simulation models can be used to predict the performance of a given user in a novel paradigm, and an example of this is shown in Chapter 5.

Although the paradigm here used stepwise integrated classifier outputs, similar analysis can be used for other integration methods. The apparent multimodality of the time-to-selection indicates that there may be a low frequency component in the trajectory of the classifier output, and this is reflected in that the models were somewhat able capture some of the secondary peaks. Recently, Saeedi et al. (2013) found that the 'fast' and 'slow' trials could be predicted by the uncertainty of the classifier before the trials began. This is useful information which could be incorporated into future models, which would seek to better model the selection time as well as the accuracy.

One caveat is that as the models were trained and tested on the same data, it is possible that there is some level of over fitting. (The decision to use all the data was due to the small number of trials, 30 for each mental class per user.) A better approach might be to train the model on a subset of runs or trials from the same session and test the predictions against the remaining runs. This would test the simulator's ability to account for the fluctuation of a user's performance in a given session. Alternatively, if data is collected which captures changes in user performance (for example, fatigue), this knowledge could be incorporated into the model as a parameter. The assumptions of stationarity may also oversimplify aspects of the output such as drift in the classifier output, and the dynamics of the control of feedback are not captured by the models. Still, the results shows that simple models are able to approximate the time and accuracy characteristics. As the actual trajectory of the feedback is what the user sees, this can be evaluated in online experiments.

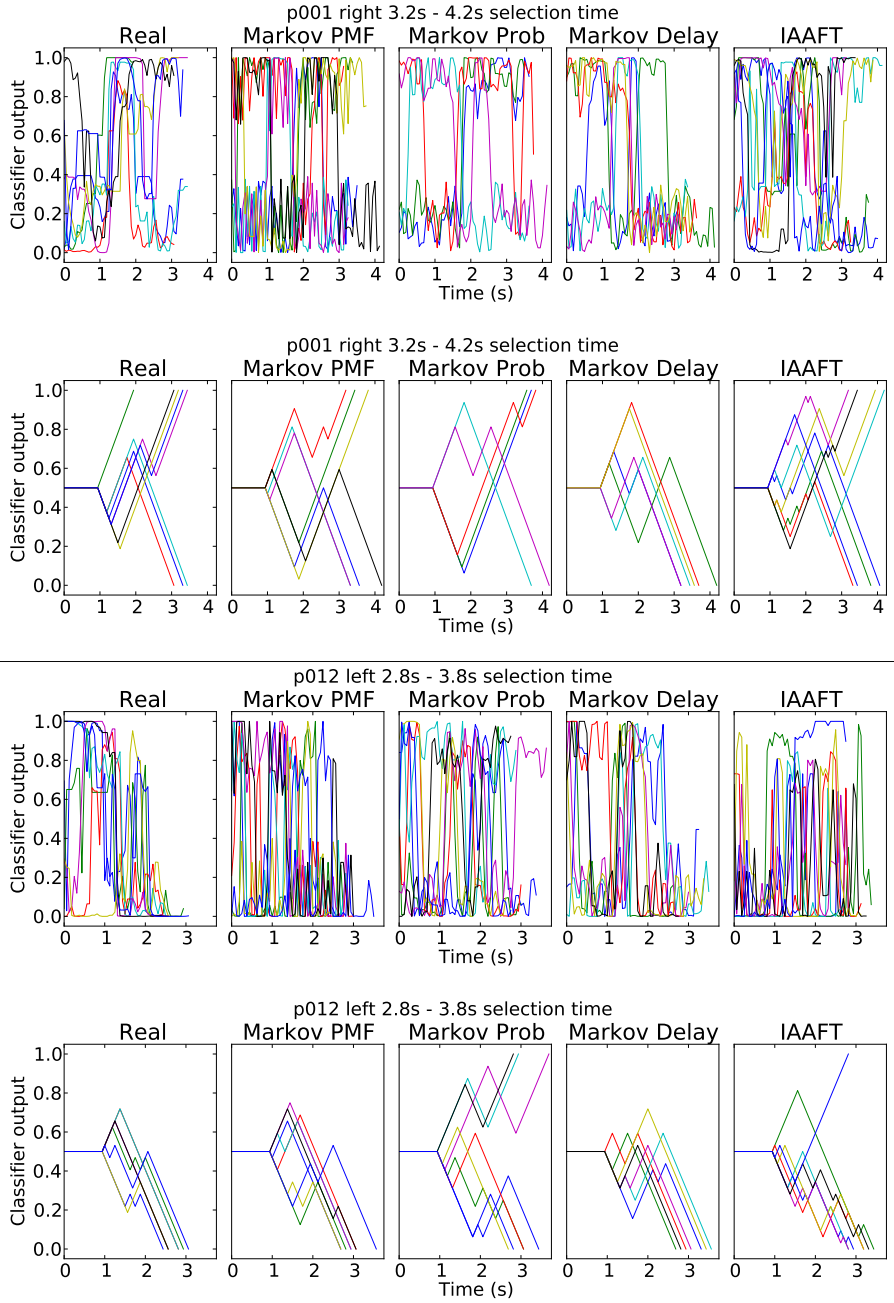


Figure 4.11: Examples of pre-integrated and integrated classifier output for two representative subjects (top and bottom). Subjectively, the Markov models match the real data better as they are less noisy than the IAAFT model.

4.6.2 Online evaluation

As the goal of the simulator is to simulate the online feel of control of the system, it must be validated by simulating the output in real time. Different levels of tests can be used to judge the quality of the simulators, with a trade-off between the cost of the test and the degree of validation afforded. Qualitative feedback and quantitative tests provide complementary information which can be used to improve the model.

Qualitative feedback can reveal aspects of the simulator that are important, for validation of the simulator, or missing, which can be used to improve it. Several variations of evaluations can be carried out in order to achieve this. In the simplest instance, BCI experts and users can simply be asked to evaluate the face validity of the simulator. This is a qualitative test for how much something looks or feels like something else (Banks et al., 2009). Participants would use the simulator, providing opinions on the similarities and differences between the simulator and real BCI. As a first pass, a 'generic' simulator that broadly captures the control characteristics of any user can be used. The characteristics picked up by participants would help to validate that the parameters of the simulator match those that users think are important in simulating the feel of control. It would also likely illuminate control characteristics that are not currently captured (or even those that could not realistically be captured) by the model.

Taking this a step further, BCI users can be asked to evaluate a simulation model whose parameters have been optimised to match their own performance. Apart from being set by the experimenter, participants could also attempt to manually tune these parameters to fit their idea of how a BCI feels like for them. In both cases, the goal would be to find out if the model's parameters are robust enough to simulate a variety of user characteristics. An added benefit of the the latter case is that this would test if the parameters are intuitive to BCI users. To provide a measure of the reproducibility of model fitting several participants could be asked to tune the parameters of the model to the same data, comparing the fit using the quantitative evaluation methods described in the previous section. Similarly, the same participant could tune the model parameters to the same data several times. The variability of the chosen parameters could provide objective measures of how precise people's perceptions of the BCI control characteristics are.

Experiences of naive users (i.e. who have never used a BCI before) can also provide valuable input which can be used to validate the simulator. For example, naive participants could be asked to describe their experiences of using a simulator, and these could be compared with BCI users' or researchers' descriptions of real BCI. This would highlight the control characteristics that are captured (or not) by the simulator. Naive users could also use the simulator prior to BCI training, then compare their experience with the simulator either through recalling their experience of using the simulator, or reusing the simulator after training. The approach would test how well the simulator captures the relevant aspects of BCI control by analysing how users' expectations of using the simulator matched up to their experience of real BCI. Quantitative tests which capture the user's experience,

such as the NASA task load index (NASA-TLX) (Hart and Staveland, 1988), could also be used to evaluate similarities and differences of using a simulator compared with real BCI. A limitation of this method is that many participants would be required to obtain enough information to observe any patterns in the data. As in all the evaluation methods, an analysis of the limitations and benefits of the simulator would be a useful result of the studies.

Evaluation using a Turing Test

The 'ultimate' quantitative test to validate a simulator is the Turing test Turing (1950). In this test, the participant is presented with a series of trials where each trial is either simulated or real. After the presentation of each trial, the participant decides if the trial was a real BCI trial or a simulated one. Again, several variations of the test can be identified. Firstly, the participant can simply watch playbacks of trials, deciding if each trial was a playback of real or simulated data. The data used can either be the participant's own, or another user's. In contrast, online trials can be carried out where the participant is actually in the loop with a real BCI. The feedback for each trial is either real (the participant is fully in control), or simulated. As before, at the end of each trial the participant is asked to assign a label to the trial. The BCI can be trained on real movements, where the participant uses an accelerometer and tries to determine if the feedback is real or simulated. The advantage of this method is that the experimenter is able to determine that the person is actually doing the task. However, it could be argued that this is not 'real' BCI in the first place as the person is actually moving. Alternatively, the experiment could be carried out on imagined movements as in a real BCI; in this case the experimenter trusts that the participant is really trying to do the mental task, rather than cheating to try to determine if the feedback is real or simulated.

To evaluate results from a Turing test, the idea is to find out if people can distinguish between the simulated and real data. Statistical tests for analysing results from a Turing test are explained in (Schruben, 1980). The author explains that the reason why classical statistical tests should not be used is that such a test is designed to guard against easily rejecting the null hypothesis that there is a difference in answers between the two (i.e. making a Type I error). This makes it easier to accept the null hypothesis that there is no difference (a Type II error). To validate a simulator, on the other hand, the goal is to guard against making a Type II error; that is, falsely accepting that there is no difference between the two when in fact there is a difference. An approach to evaluating the results of the test is then proposed, which adopts a Bayesian approach. Yet, the proposed method of analysis takes into account only the number of simulated trials that are judged as real or simulated, without incorporating the number of real trials that were judged to be simulated. An alternative approach is to adopt a Bayesian approach to model selection (Bishop, 2006).[1] This calculates a probability indicating which of 2 competing hypotheses or models is more likely given the available evidence. Using Bayes' theorem, the posterior probability of a

[1]Thanks to Simon Rogers for help with the method and equations.

model M given the data D is proportional to the likelihood $\times$ prior:

$$P(M|D) \propto P(D|M) \times P(M)$$

For two models with equal probability (prior) M_i and M_j, the probability of M_i is

$$p(M_i|D) = \frac{p(D|M_i)}{p(D|M_i) + p(D|M_j)}$$

In the current case, if N_r is the number of real trials and N_s is the number of simulated trials, the data D is the sum of n_{rr}, the number of real trials correctly judged to be real, and n_{sr}, the number of simulated trials incorrectly judged to be real. The competing models (hypotheses) can then be defined as:

- Model 1 (M_1) - the same mental model was used to judge between the simulated and real trials (i.e. there was no difference in the way a trial was judged regardless of being simulated or real). As this implies a single model parameter, y, the evidence can be computed as

$$\begin{aligned} p(D|M_1) &= p(n_{rr}, n_{sr}|N_r, N_s, M_1) \\ &= \int p(n_{rr}, n_{sr}, y|N_r, N_s, M_1)dm \\ &= \int p(n_{rr}|N_r, y, M_1)p(n_{sr}|N_s, y, M_1)p(y|M_1)dy \end{aligned}$$

- Model 2 (M_2) - different mental models were used to judge between the simulated and real trials. In this case, two parameters y_1 and y_2 are used and the evidence is computed as

$$\begin{aligned} p(D|M_2) &= p(n_{rr}, n_{sr}|N_r, N_s, M_2) \\ &= \int\int p(n_{rr}, n_{sr}, y_1, y_2|N_r, N_s, M_2)dy_1, dy_2 \\ &= \int\int p(n_{rr}|N_r, y_1)p(y_1|M_2)p(n_{sr}|N_s, y_2)p(y_2|M_2)dy_1, dy_2 \end{aligned}$$

Thus, the evidence is computed by integrating the likelihood of the data over the parameters of the model. Closed-form solutions for the evidence can be found using a Beta-Binomial model, where the Binomial distribution

$$L(n|N, y) = \binom{N}{n} y^n (1-y)^{(N-n)}$$

calculates the probability of the data given the model parameters, and the Beta distribution

$$\pi(y; \alpha, \beta) = \frac{1}{\mathrm{B}(\alpha, \beta)} y^{\alpha-1}(1-y)^{\beta-1}$$

marginalises the parameters over a prior distribution which is uniform when $\alpha = \beta = 1$. The probability that different mental models were used to judge simulated and real trials is then computed as

$$p(M_2|D) = \frac{p(D|M_2)}{p(D|M_1) + p(D|M_2)}$$

In this approach, the simplest model that can explain the data is favoured over more complex models, such that the probability of M_1 is higher if n_{rr} and n_{sr} are close. On the other hand, the model is also sensitive to differences between n_{rr} and n_{sr}, especially as the number of trials increases, since M_2 would be better able to account for the difference. Thus if there are large differences, M_2 is preferred over M_1 and in this way there is no bias towards M_1. Looking at it another way, there is always some chance that M_2 is correct even if n_{rr} and n_{sr} are close, while if they are far apart M_2 explains the data better; thus the probability of M_2 never reaches 0 but increases towards 1 faster if there is a difference.

Experiments

2 pilot experiments have been carried out with participants being shown playbacks of trials and being asked to guess which were simulated and real. One experiment was carried out with BCI researchers and the other with real participants. As this is a novel experiment for BCI, it was necessary to compare a plausible (as judged by the author) simulator to one that was thought to be a 'bad' simulator. In the case that there is no difference between the 'good' simulator and the real data (participants use the same model for both simulators), this is a check that participants can at least separate out trials that are assumed to be sufficiently different to real data, rather than either randomly guessing or biasing decisions in a particular direction. The hypothesis is that there will be insufficient evidence of a difference between the 'good' simulator and the real data, but that there will be a difference between the 'bad' simulator and real data. The trials for the 'bad' simulator are taken to be the baseline for which participants, if they are doing it correctly, should be able to tell are simulated trials.

In both experiments, the parameters for the 'bad' simulator were set by manually tweaking the parameters such that the integrated classifier output would produce a large number of oscillations before reaching the target. The parameter calibration for the good simulator are described in the respective experiment method sections. Trials from the bad, good and real data were stored and presented to the participants in random order. After each trial that was played back to the participant, they indicated using the left or right shift keys whether they thought the trial was simulated or real. After the experiment, participants were asked to describe the criteria they used to tell the difference between real and simulated trials.

Experiment 1: Turing test with BCI researchers

Method. Participants were 10 BCI researchers from the same lab that developed the BCI system, which also provided the BCI participant's data that was used for the simulator.[2] Participants viewed feedback trials from the BCI participant as if they were standing behind the BCI participant and watching him carry out the calibration trials. The trials were either generated from the BCI participant's real data, data from the 'good simulator', or data from the 'bad simulator'. All participants were informed of who the BCI participant was whose data was being played back, that this was a day where he had fairly good performance, and were at least somewhat familiar with the person's BCI performance and behaviour. Selection accuracy for the left trials was 64%, 100% and 68% for real data, bad simulator and good simulator respectively, and 94%, 95% and 89% for the right trials. Figure 4.12(a) shows examples of integrated classifier output of real trials, and data produced by the two simulators. The real data look comparable to the good simulator, while the bad simulator is clearly different with oscillations around 0.5. Figure 4.12(b) compares the time-to-selection of the trials. Although comparable, the real data appears to have a longer tail than the good simulator. The bad simulator has a vastly different distribution of timings than the real data, in most cases either taking a much longer time ($\sim$70 seconds) or a much shorter time (less than 1 second) to reach the threshold. 5 left and 5 right trials for each data type were presented in randomised order, for a total of 5 trials $\times$ 2 targets $\times$ 3 simulators = 30 trials. The integration method used was the exponential smoothing function as in during the data collection, and the *Markov delay* model was used for the 'good' simulator. Parameters were selected by running the optimization algorithm a number of times and selecting the set that produced the most reasonable simulator both offline and online. Since the experiment was done remotely, participants were asked to provide a typed description of their criteria used to judge the trials, as well as any comments about the experiment.

Results and discussion. Figure 4.13 shows the scores and the probability that different mental models were used to judge the trials ($P(M_2|D)$). The mean and standard deviations for the 'bad' simulator, 'good' simulator and real data were 3.4 ($\pm$ 2.54), 6.4 ($\pm$ 1.8) and 5.7 ($\pm$ 1.45) respectively. Data from 7 out of the 10 participants follows a trend such that the number of trials marked 'real' for the 'bad' simulator was $\leq$4, while the number for the 'good' simulator and the real data were comparable at $\geq$5. In all of the cases where there was a high probability ($\geq$0.8) of model 2, this was due to a difference either between the bad simulator and the real data (p6, p8 and p16), or the bad simulator and the good simulator (p2, p6, p14, p16, p19). It is interesting to note that on average, fewer trials for the real data were marked as 'real' than for the 'good' simulator. Subjective feedback from participants as to how they judged the trials can help to illuminate this finding.

Participants' criteria for scoring trials as real included consistent movement towards the correct target (p2, p4, p16), 'smooth movements' in general (p16), smooth movements towards the end of the trial (p8), and oscillations in the wrong direction 'some of the time

[2]While the experimental design and protocol was done by the author, the experiment was carried out by Serafiem Perdikis from EPFL

but not too often' (p2, p4). p14 indicated that a trial was marked real when there were 'strange oscillatory behaviors, followed by smooth increase until the threshold'. 7 out of 10 participants mentioned the nature of the oscillations as criteria for judging a trial to be real or simulated. The main criteria used to judge if a trial was simulated appears to be large, inconsistent oscillations, or if the bar moved 'too much' or too quickly. 4 participants indicated marking a trial as simulated if it took too long to reach the threshold. One participant (p4) indicated that trials that were 'very smooth' or or 'very sluggish' were judged to be simulated. Big jumps in the trial, a feature associated with the 'bad' simulator, were also marked as simulated (p8, p16).

Some of the criteria used to judge trials as being simulated may have led to incorrect labelling of real trials. For example, p6 wrote, "Whenever there were oscillations, I clicked simulator option". This may partly explain the somewhat conservative marking of trials as being real (6/10 for real trials). Generalisation of the features of BCI control may also have led to this. p19 stated, 'if it approaches quickly to the correct trial zone without slowing down when approaching the zone, I feel also that trial as fake'. Although in some cases this may be true, it is not clear that every BCI trial should slow down when reaching the threshold. Underestimating the time taken to reach the threshold for some trials may also have led to incorrect labelling of the real trials, consistent with p2's statement, 'whenever the trial was taking too long to finish, I would also consider the trial fake, because my experience says that a real subject will manage to reach the correct target within the first 5-8 seconds, otherwise he/she will get too tired (especially if he is biased) and give up'. Yet there were some trials that lasted longer than 10 seconds, (Figure 4.12), indicating that in some situations the participant would struggle for longer than the time expected by p2.

2 participants explicitly said that in general, they had difficulty telling the difference between the simulated and real trials. p2 wrote 'I was most of the times in doubt whether my selection was correct)'. 3 participants (p2, p10, p14) indicated that knowing more about the characteristics of the feedback for the subject would have helped to judge the trials. This indicates that BCI researchers have a strong sense of there being differences in BCI characteristics in terms of the behaviour of the feedback bar.

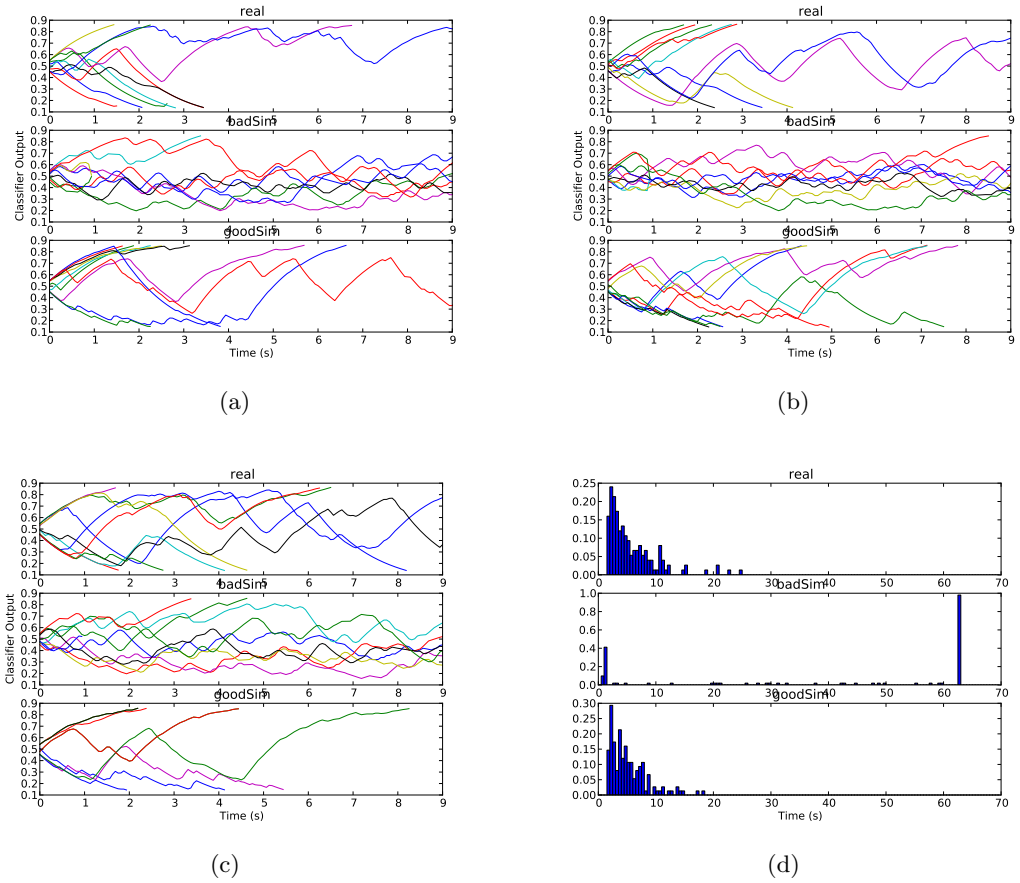
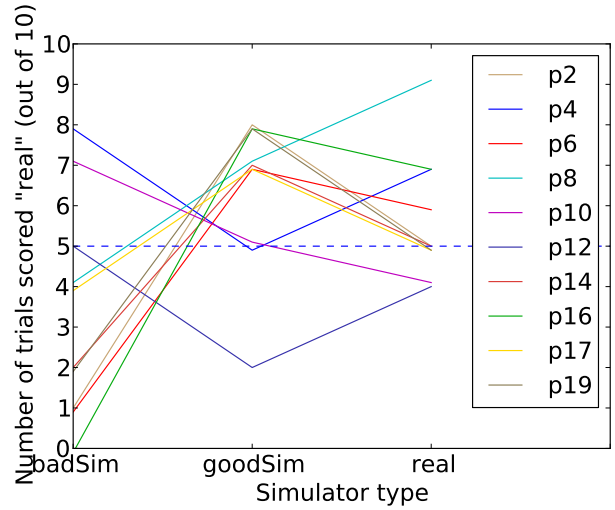
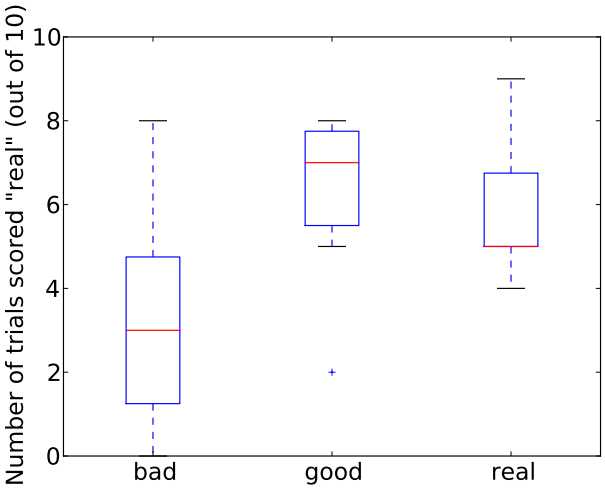


Figure 4.12: (a)-(c) Random samples of the time series of the cursor feedback as seen by the BCI experts in experiment 1, 10 trials for each of real data, 'bad' simulator and 'good' simulator. (d) Histogram of time-to-selection (TTS) for the trials.

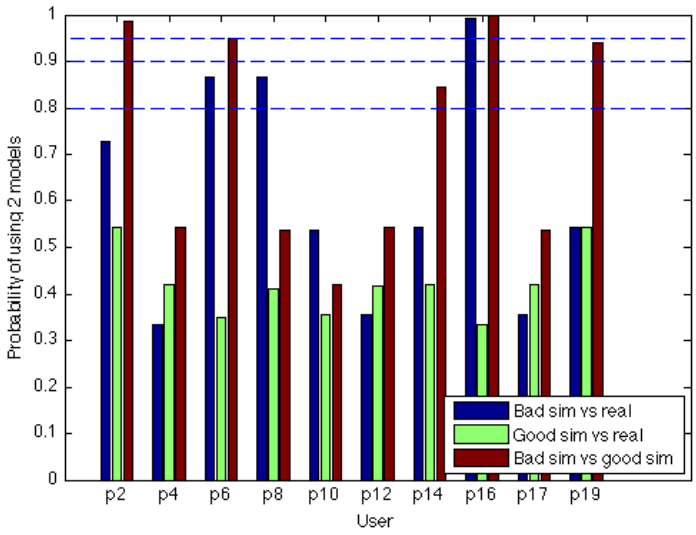


(a)

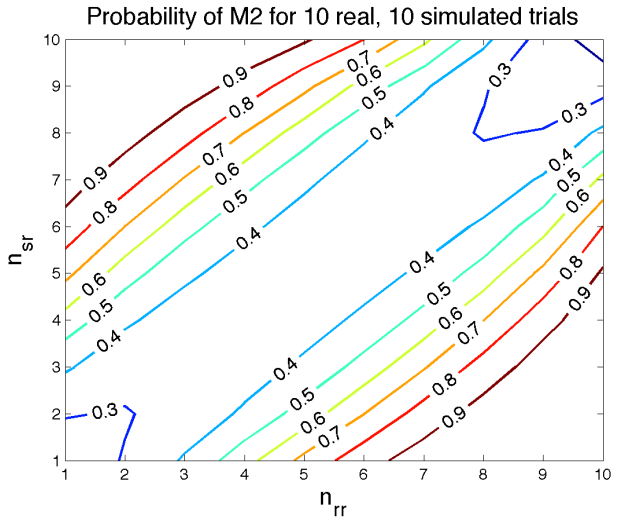


(b)

Figure 4.13: Turing test scores for BCI experts judging playbacks of trials for one BCI subject. (a) Number of trials scored 'real' for the 'bad' simulator, 'good' simulator and real data. (b) Boxplot of scores over all 10 participants.



(a)



(b)

Figure 4.14: Results of Turing test experiment where BCI experts judged playbacks of trials for one BCI participant. The probability of a participant using the same mental model to evaluate all the trials is denoted M_1, while the probability that two different models were used to evaluate trials from two different sources is denoted M_2. ($p(M_1) = 1 - p(M_2)$.) (a) Probability of M_2 for bad simulator vs real, good simulator vs real and bad simulator vs good simulator. (b) Contour plot showing the probability of M_2 for different values of n_{rr} (actual real, answered 'real') and n_{sr} (actual simulated, answered 'real'). Blue indicates that the data favours M_1, while red indicates that M_2 is very likely.

Experiment 2: Turing test with healthy participants

Method. 4 people who had undergone MI training took part in the experiment. As they were in the lab for another experiment, participants did the MI calibration trials first as a standard warm-up or calibration to ensure that performance for the day was adequate for the experiment. While they did the actual experiment, data from the calibration trials were used to create a simulator. At the end of the experiment, participants were then asked to do the simulator experiment. In order not to confuse participants, the trials for the left and right targets were separated. The experiment thus consisted of 15 trials of each of the bad simulator, good simulator and real data for the left target, in randomized order, followed by 15 of each for the right target, giving a total of 15 trials $\times$ 3 simulators $\times$ 2 targets = 90 trials.

In this experiment, the step integrator was used as the evidence accumulation. For p005 and p012, the model and parameter tuning method was the same as the previous experiment, while for p026 and p027 the *Markov Pmf* model was used to calibrate the parameters. The time series of the integrated classifier output that was seen by participants was recorded for participants p026 and p027 (but not for p005 and p012). Upon viewing the recorded data, it was found that a bug in the experiment program caused the data to sometimes take a longer time to reach a threshold than the actual recorded time series. This was due to a change in the calibration trials program prior to carrying out the simulation experiment for these last two participants, which was not taken into account in the program for the simulation experiment. This may have had a negative impact on the validity of the experiment; nevertheless as their data and subjective feedback are still useful, the results are reported here.

Results and discussion. Results for the data for participants is shown in Figure 4.15. For 3 out of 4 of the participants, the probability that the trials for the bad simulator could be distinguished from real trials was high (> 0.8) for both left and right trials, and for 6 out of 8 data points the probability of model 2 was equally high. p005 rated most of the trials as real. In terms of subjective experiences, p005 explained that he thought all of the trials were plausible, as while he was doing the motor imagery tasks, he was concentrating on the tasks and not really focussing on the movement of the cursor. p012 found it a struggle, saying that in many cases he did not know whether his answer was correct. p026 found that he thought that around a third of the trials were simulated, while p027 found the task 'ok' in terms of difficulty. Noisy behaviour, or 'jitter', was reported by p012 and p026 as the main criterion for distinguishing between real and simulated trials, while p027 indicated that as he could not remember any trials where there was more than one turning point, he marked all the trials where this happened as simulated.

Since, as previously mentioned, there was a bug in the experiment program for p026 and p027, a detailed case analysis for each participant is deserved. In each case, the effect of the bug was that some of the trials did not reach the threshold at the recorded time, and were thus longer than the recorded trials. Figures 4.16–4.19 show the time series of trials that

participants actually saw and how they judged them (top) and the recorded trials for the real data and good simulator (bottom). Figures 4.16 and 4.17 show that for p026, the trials that were presented to the participant as real and those for the good simulator were longer and noisier than the trials that were actually recorded. However, the participant marked a fair number of these longer trials as real, indicating that to some extent, he was willing to accept longer and noisier trials as being real.

For participant p027 (Figures 4.18 and 4.19), it can be seen that in the instances where the trials were comparatively long and there was more than one turning point, the participant marked these as simulated, as he reported. This occurred for both the good simulator and the real data. A comparison of the difference in time series between trials seen by the participant and the trials recorded confirms the presence of this bug in the experiment code, and two interesting points can be made from the results. Firstly, the trials for 'goodSim wrong' and 'real correct' (trials judged to be real) show similar characteristics in terms of the time taken to reach the threshold, while the trials for 'goodSim correct' and 'real wrong' (trials judged to be simulated) also shared similar characteristics in that there was more than one turning point in each of the trials. Trials for bad simulator vs good simulator or real data were consistently distinguished from one another, while the probability of M_2 for good simulator vs real trials was consistently low. This shows that the participant had a very specific mental model of his own perceived performance. That the time series of the recorded data for both real and simulated data appear to have fairly similar characteristics strengthens support for the validation of the simulator, showing that the simulator did not generate trials that were markedly different from the real data. Secondly, it is interesting that p027 indicated that he could not remember trials where there was more than one turning point, as the real data shows 4 instances where this actually happened. While this is a very small number of trials (4/60 or 6.7%), this indicates that he might have marked these trials as simulated if he had actually seen them. Finally, although the participant judged the trials by the number of turning points, it would appear that he did not take into account the time taken for the turning points or remember the target for which they occurred (there were no real trials that had a turning point for the right target, yet these were what were presented in the simulation experiment and which he actually accepted as being real).

The discrepancy in performance between p005 and the other participants in distinguishing the bad simulator from the good simulator and real data is interesting as it supports the need to have carried out the experiment with the bad simulator in the first place. (If all the participants could not tell the difference between real and simulated data, one would question whether people generally notice difference between trials at all.) It is unclear why this participant did not share the same pattern of scores with the other participants. If his performance had been poorer than the other participants' (trials were noisier and took longer), a plausible explanation would have been that as the trials for the bad simulator were similar to the real data, they were easily confused with the trials for the good simulator. However, this did not seem to be the case as the scores for all the participants were high.

It is possible that p005 was simply tired that day or was judging the trials as he thought were plausibly real BCI trials rather than being his own trials. Similarly he might have been judging the trials according to his past performances as a whole rather than at the time of the trials, or he may have forgotten what his performance was like in between doing the calibration trials and the simulation experiment. It may also be that he simply did not pay attention to the characteristics of the trials, or forgot what the trials were like, which would be in line with what he reported.

General Discussion

The experiments showed that participants' reported criteria for judging if a trial was simulated or real was fairly consistent across the BCI researcher and BCI participant groups, with the main criterion being the degree of oscillations in the feedback bar. A few participants also mentioned taking into account their expectation of accuracy of the feedback bar moving towards the correct target. Across most participants in both experiments, a similar pattern was followed such that the probability of M_2, that different mental models were used to judge between simulated and real trials, was high (> 0.8) for many of the data points in judging between the bad simulator and either real data or the good simulator, while the probability of M_2 for distinguishing between the good simulator and real data was consistently < 0.6 for experiment 1 and < 0.5 for experiment 2. The slightly higher probabilities observed in experiment 1 can be accounted for by the smaller number of trials used in experiment 1.

It is interesting that for most participants, some of the real trials were judged to be simulated. There are a few ways this could have occurred. Firstly, it could have been due to random guessing, where the participant was unsure whether the trial was simulated or real. Secondly, it could have been due to a bug in the system, as was with p026 and p027 in the second experiment, where the trials seen by the participants were longer than the actual trials recorded due to a sampling error. Thirdly, incorrect labelling could have been due to an incorrect assumption about the real data such that some feature of the real trial led to the incorrect conclusion that the trial was simulated. This clearly happened for some of the BCI experts in experiment 1, which could be due to them not having witnessed the performance of the BCI participant prior to the experiment. Where it happens for participants viewing their own data versus simulated data, this would mean that the model of the user's own performance was somewhat different from reality. Underestimation of one's performance could lead to rejecting short trials or trials that correctly hit the target, while overestimation could lead to rejecting longer trials or trials that reached the wrong target. However, the amount and quality of data in this experiment is insufficient to draw conclusions about how often this would happen, when, and if there are individuals who perform better on such tests. Further experiments should be carried out to identify how much people are actually aware of their own performance; the Turing test methodology presented here provides a basis for carrying out such experiments.

Where simulated trials were correctly labelled as simulated, this could again happen by

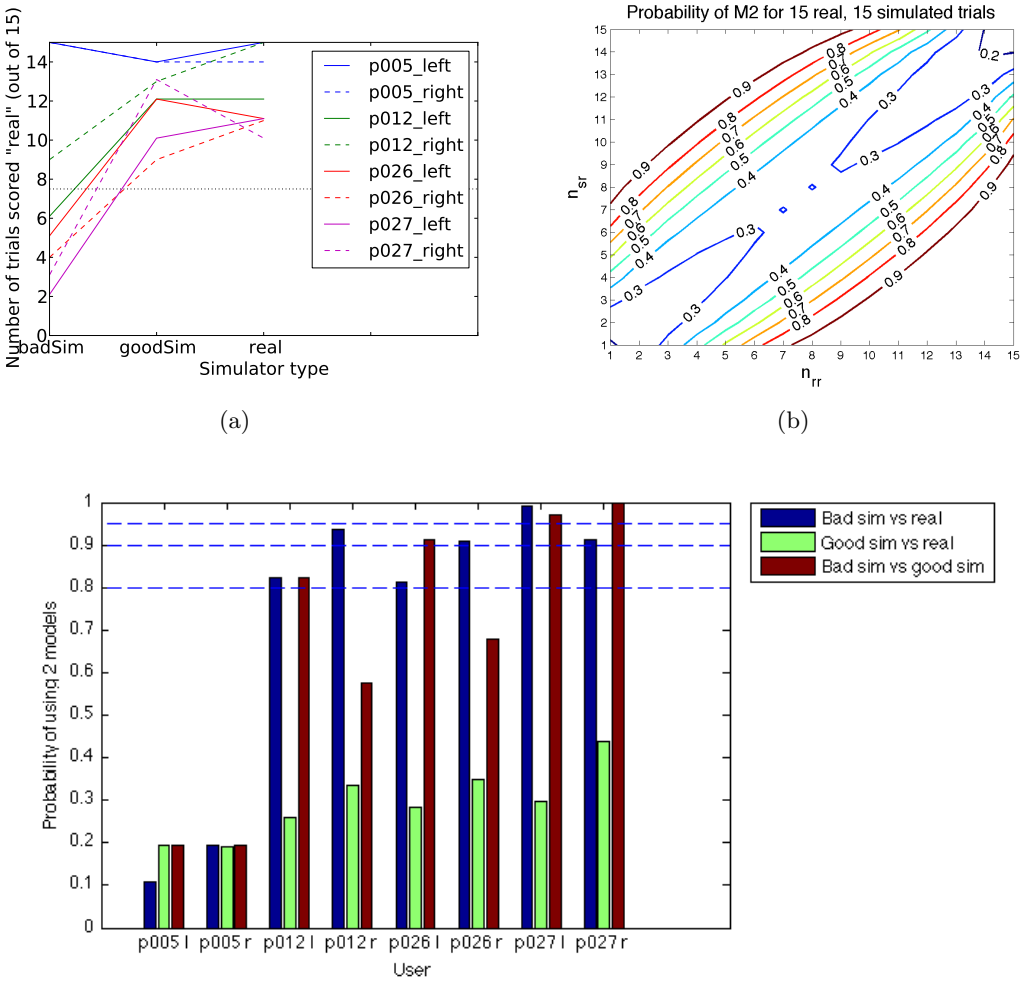


Figure 4.15: Turing test scores for participants judging playbacks of their own trials after an experiment, for left and right trials. The probability of a participant using the same mental model to evaluate all the trials is denoted M_1, while the probability that two different models were used to evaluate trials from two different sources is denoted M_2. ($p(M_1) = 1 - p(M_2)$.) (a) Number of trials scored 'real' for the 'bad' simulator, 'good' simulator and real data. (b) Contour plot showing the probability of M_2 for different values of n_{rr} (actual real, answered 'real') and n_{sr} (actual simulated, answered 'real'). Blue indicates that the data favours M_1, while red indicates that M_2 is very likely.(c) Probability of M_2 for bad simulator vs real, good simulator vs real and bad simulator vs good simulator.

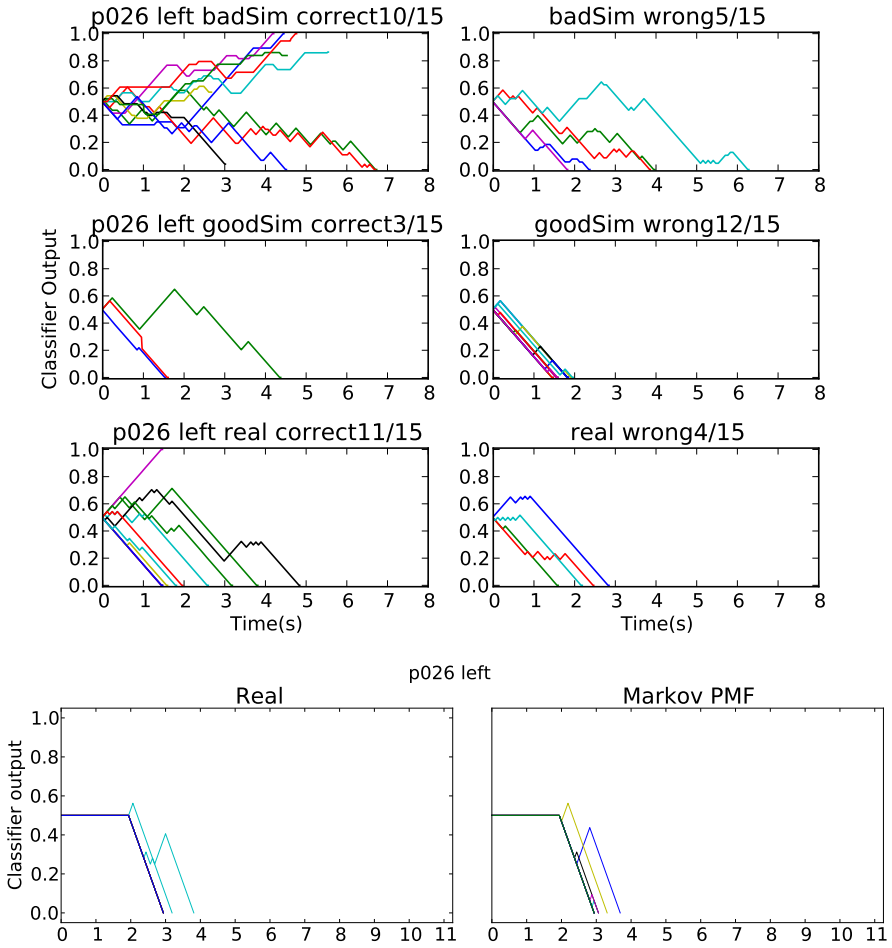


Figure 4.16: Feedback display (linear-integrated classifier output) of online evaluation experiment for p026, left trials. The top graphs show the actual feedback trajectories seen by the participant, while the bottom graphs show the actual recorded data for real trials and simulated trials ('good' simulator trials).

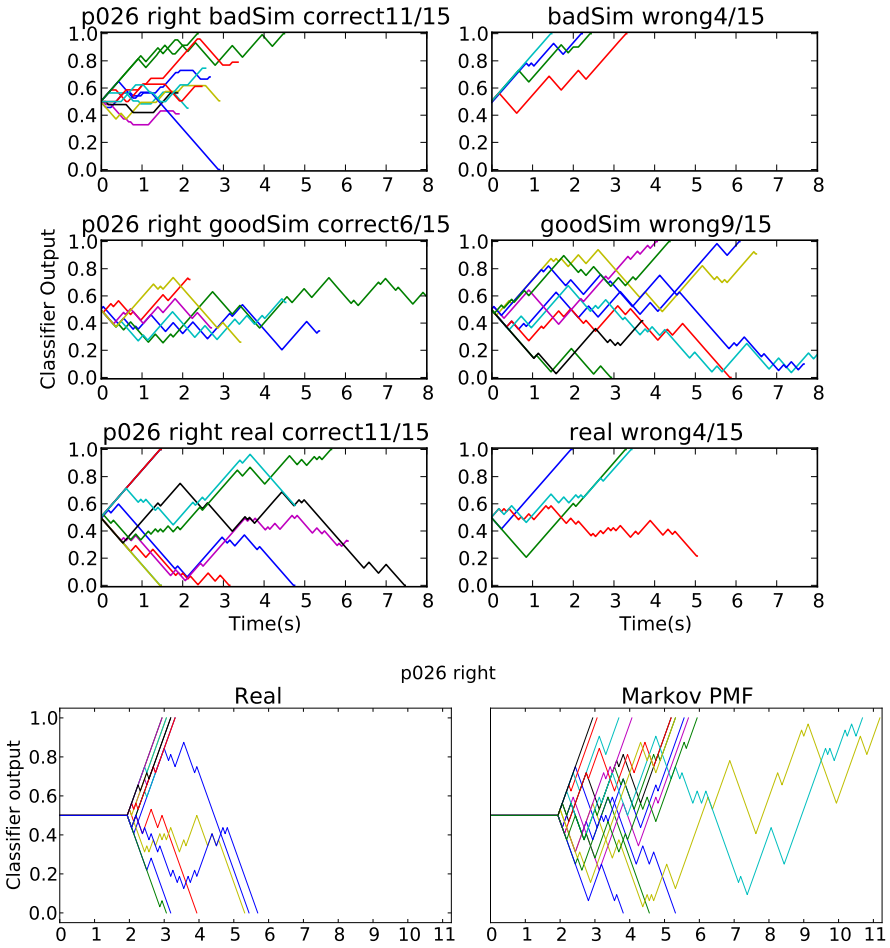


Figure 4.17: Feedback display (linear-integrated classifier output) of online evaluation experiment for p026, right trials. The top graphs show the actual feedback trajectories seen by the participant, while the bottom graphs show the actual recorded data for real trials and simulated trials ('good' simulator trials).

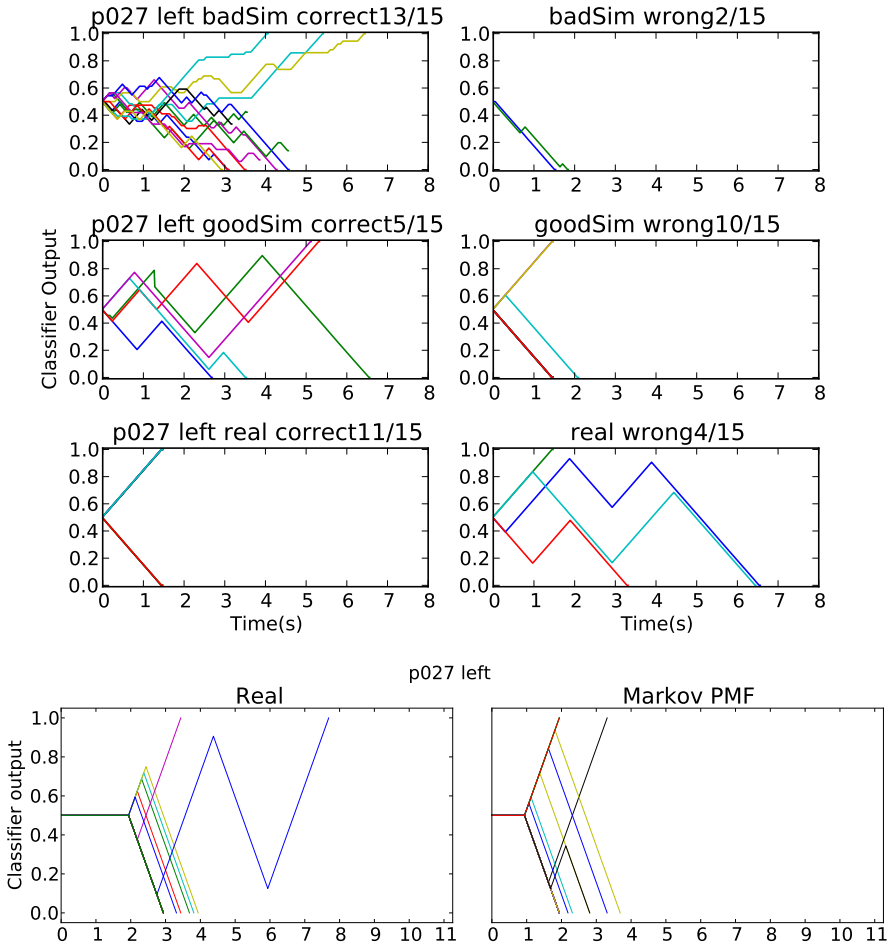


Figure 4.18: Feedback display (linear-integrated classifier output) of online evaluation experiment for p027, left trials. The top graphs show the actual feedback trajectories seen by the participant, while the bottom graphs show the actual recorded data for real trials and simulated trials ('good' simulator trials).

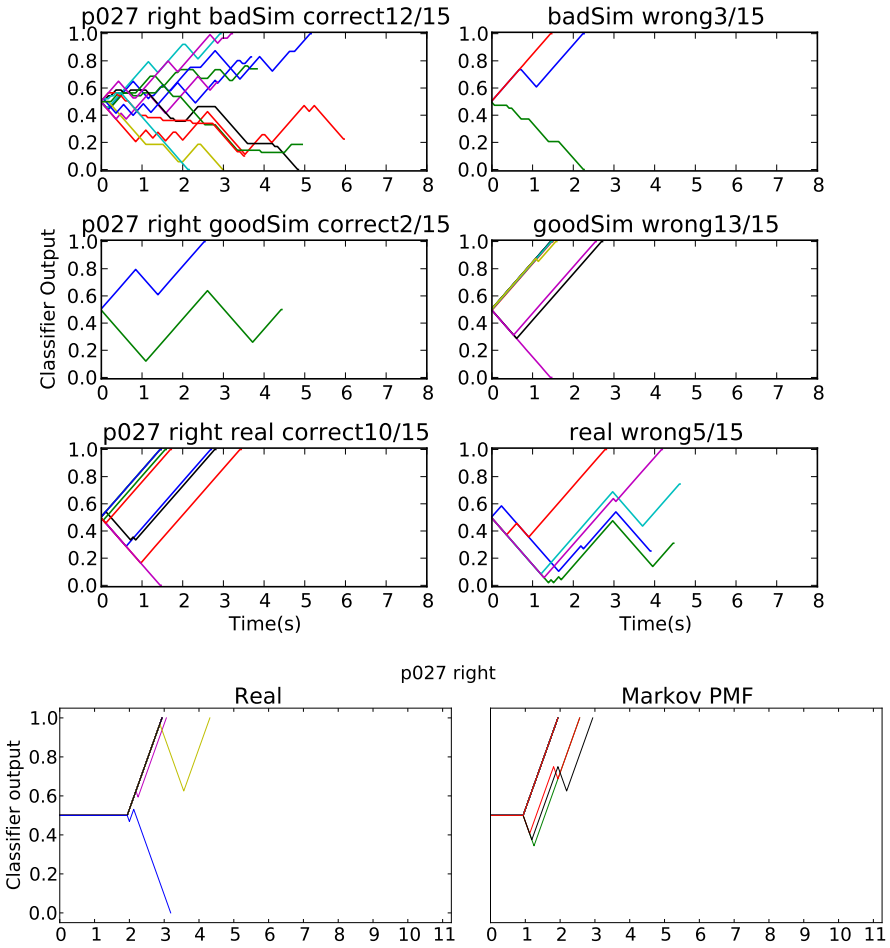


Figure 4.19: Feedback display (linear-integrated classifier output) of online evaluation experiment for p027, right trials. The top graphs show the actual feedback trajectories seen by the participant, while the bottom graphs show the actual recorded data for real trials and simulated trials ('good' simulator trials).

random guessing, and similarly, trials that correctly simulated the characteristics of real data could have been misplayed due to the experiment bug. A 'correctly' simulated trial could also have been misjudged by the participant due to an incorrect assumption or mental model of the characteristics of the real trials. Even if the trial characteristics were the same as a real trial, a participant having a good memory may have spotted that the simulated trial was not actually one that they had seen before, thus marking the trial as simulated not because it did not reflect the right characteristics, but simply because it was not a real trial. This did not appear to be an issue in the experiment, as the recordings for p026 and p027 showed that even trials that were not exactly the same as real trials were marked as real, indicating that participants do not remember the exact trajectory of the feedback. Finally, the participant may have judged the trial to be simulated due to a correct judgment that the trial had different characteristics to a real BCI trial. It is of course possible that while the majority of trials produced by the real simulator are closely matched to the real trials, several trials could display deviant characteristics from the real trials. This is a reasonable expectation of the performance of the simulator, as a model is generally always an approximation of the real system (Sargent, 2010). Conversely, a higher proportion of simulated trials being labelled as real than the real trials could mean that the simulator did not sufficiently capture all the characteristics of the real data.

Thus, the attempt to validate the simulator with the Turing test raised several interesting questions with respect to how people perceive or remember performance in an MI-BCI. Firstly, users' main criterion for distinguishing between real and simulated trials was the degree of noise in the feedback. This suggests that to some extent, people pay attention to the trajectory of the feedback as well as the performance criteria such as selection accuracy and time to reach the target. The interaction between these aspects may influence the *feel* of, or *perceived*, control of the interaction. Further investigating the relationship between these aspects would be useful for the design of BCI feedback, since it is widely accepted by now that users' perception of performance is a more important criterion for acceptability of user interfaces than actual performance. For example, Harrison (2007) found that changing the acceleration of a progress bar from left to right while keeping the time constant affected user's perceptions of how quickly the bar reached the end point. The perception of the passage of time is linked to the sense of agency, or the feel of being in control of an outcome (Ebert and Wegner, 2010); overestimation of elapsed time is linked to low levels of perceived agency, while underestimation is linked to a high level of perceived agency. Lynn et al. (2010) found that people's perceived intent to control movements of a line via a simulated BCI could be manipulated by the number of times the line actually moved. Thus far, there have been no studies investigating how the feel of control or perception of performance is affected by the speed or level of noise of the feedback. It is possible that manipulating these aspects can improve the perception of control for BCI users, thus improving the user experience. The simulator could be a useful tool to carry out these investigations.

Finally, the current experiment methodology (calibration trials) used to investigate the feel of control of a BCI did not allow us to capture users' perceived delay of mental task

switching. To do this, a continuous experiment could be designed such that the user imagines one class for a period of time, and switches to another class at a set time. One issue is that it is difficult to pinpoint exactly when the user switched mental states, and when the feedback actually moved in response to the user's change in mental state. Another is that the signal within a mental state is itself corrupted by noise. A workaround would be to ignore all the trials that were wrong, i.e. where the user did not manage to reach the correct target. The issue of interest is then how the user's perception, that is the feel of delay, relates to the actual delay.

4.7 Conclusions

A conceptual model of factors which influence the feel of control of a MI BCI, the first of its kind, was developed and validated in conjunction with BCI researchers. Possible metrics that can be used to quantify the time series of a BCI signal, with regard to simulating the feel of control, were identified. The implemented approaches to simulating the signals were described, and some methods of evaluation were discussed. Preliminary results indicate that the current simulator models are able to produce characteristics that are similar to the real datasets in terms of selection accuracy, time-to-selection and classifier output time series, and are similarly confused by participants viewing playbacks of the simulated and real feedback trials. Of the BCI researchers who tried to detect the difference between real and simulated trials in an online Turing test, there was evidence to suggest that around half could distinguish the trials generated by the bad simulator from either the good simulator or the real data, but not between the good simulator and the real data on which it was modelled. More experiments are required to validate the models for individual participants and to ascertain the nature of how the signal features map to the feel of control (i.e. what features lead to users' belief that the presented feedback is real). However, the generic simulator may be considered to be a good enough approximation to a real BCI for the purpose of design and development.

The goals set out at the beginning of the chapter for building a simulator were to capture the feel of control of a BCI for use in design and for communicating with stakeholders. The current simulator models have been developed to capture at least the oscillatory behaviour, speed and accuracy of MI-BCI trials. Further improvements to the model and experiments can help to ascertain the extent to which these models reflect the real BCI, and to what extent people perceive, and the simulator can capture, individual characteristics or classes of characteristics representing groups of individuals. However, the current models represent the first steps towards the goal. In terms of development and debugging, the described simulator can be useful in providing classifier output values that can be directly used in a BCI application. The usefulness of the simulator in developing a novel control paradigm or an application is demonstrated in the online simulation user studies described in Chapter 6. In addition, it may be possible to use the simulator to generate offline predictions of task performance in a novel paradigm. This is demonstrated in the next chapter.

5 Applications I: the REx (Rotate-Extend) Paradigm

Summary. This chapter provides an example of using the simulator models developed in Chapter 4 to predict task performance in a novel BCI paradigm using offline simulation. Simulation predictions based on binary data and actual results were compared for 10 participants in a paradigm named the Rotate-Extend (REx) paradigm. The main findings are that simulation can be useful as an offline tool for predicting user performance. Using different models can help to strengthen the quality of the predictions as they capture different aspects of user behaviour.

5.1 Introduction

The binary paradigm for a 2-class Motor-Imagery Brain-Computer Interface (MI-BCI) as discussed in previous chapters is a natural design choice for interaction as it directly follows the offline training task paradigm. Nevertheless, it is not evident that the mental classes used for training in the offline task should be the same as that for an actual application. There is also no inherent reason why the interaction should be discrete, as long as there are no psychological or neurophysiological difficulties arising from continuous control or imagination of the mental class. Several examples of control paradigms for 2-class MI-BCIs which are alternatives to the discrete binary paradigm have been reported in the literature. In Yue et al. (2012)'s study, 5 out of 6 participants were able to continuously balance a simulated inverted pendulum for more than 35 seconds; Tonin et al. (2011) describes a telepresence robot controlled by using a 2-class MI-BCI to turn it left or right by selecting the corresponding targets, and keeping it moving straight ahead by keeping the cursor in between targets; Scherer et al. (2008) similarly describes a user moving around in a virtual apartment; the ubiquitous Brain Pong game (Kerepki et al., 2007) assumes continuous control of the horizontal position of a paddle on the screen. McFarland et al. (2003) explored participants' ability to select a target out of 2, 3, 4 or 5 targets by controlling the vertical position of the ball, while Friedrich et al. (2009) investigated the use of a scanning mode where a relaxed mental state advanced a cursor through 4 sequential targets, and a MI class was used to select the target. In these tasks, considering the dynamics of control is important, as the tasks depend on the user's ability to time the production of their mental states to coincide with the correct movement of the relevant interface object on the screen. The tasks mentioned above represent both continuous *time* control, where 'continuous' here

refers to the user's perception of continuous motion, and continuous *state* control where the classifier output is not simply a set of discrete classes but a real number.

In most of the studies considered above, an offline training phase is followed by an online training phase with discrete calibration trials. The desired control paradigm is then tested, either by adaptive algorithms or with the researcher setting the parameters for the participant. A useful tool would allow a designer, researcher or practitioner to gauge how well a particular participant might perform for different paradigms, and to optimize the parameters of the paradigm for the participant. McFarland and Wolpaw (2003) showed that simulation of task performance using a simple model could predict the trends in performance, such that changing the gain of cursor movement predicted which targets would be more easily selected given different levels of the gain parameter (high, medium or low). However, the gain parameter that would lead to optimal performance requires to be fine-tuned for each individual. It is also possible that individual characteristics can mean that the best control paradigm is different for different participants. If this knowledge can be ascertained without having the participant experiment with many parameters and paradigms, time and effort required to manually select the best options could be saved. In this chapter, the Rotate-Extend (REx) paradigm, a generalisation of the Hex-o-spell (Blankertz et al., 2006b; Williamson et al., 2009) paradigm is investigated with a view to finding out if simulations which utilise the knowledge of individual control characteristics can be used to predict individual task performance. In particular, the aim is to find out if it is possible to generalise from the binary selection task to a more complex paradigm. It is expected that models of the binary calibration data can be used to make predictions of task performance for individual users in the REx paradigm; specifically, that an individual user's performance for the REx paradigm will fall within a 95% prediction interval of simulated performance in terms of selection accuracy and task time.

5.2 REx paradigm and parameters

The Rotate-Extend (REx) paradigm consists of a wheel divided into *segments* (Figure 5.1). An arrow in the centre of the wheel controls the selection of target segments. One mental class is used to control the rotation of the arrow, and the other class extends the arrow to select the target segment. A feedback bar may be displayed, which displays the integrated classifier output in a similar manner to the binary feedback. In contrast to the original Hex-o-spell program, which had a fixed number of segments, the REx paradigm has a variable number of selections. Although the language model in the original Hex-o-spell allowed the placement of the arrow such that overall the first few segments were most often selected, in the REx paradigm the assumption is that each segment is selected with equal probability. Thus, it is unclear what the optimal combination of speed and number of segments in the wheel is that would enable the user to achieve the highest bit rate.

Table 5.1 provides a summary of the REx parameters and the initial values set. The MI classes used for rotation and extension of the arrow were assigned according to user

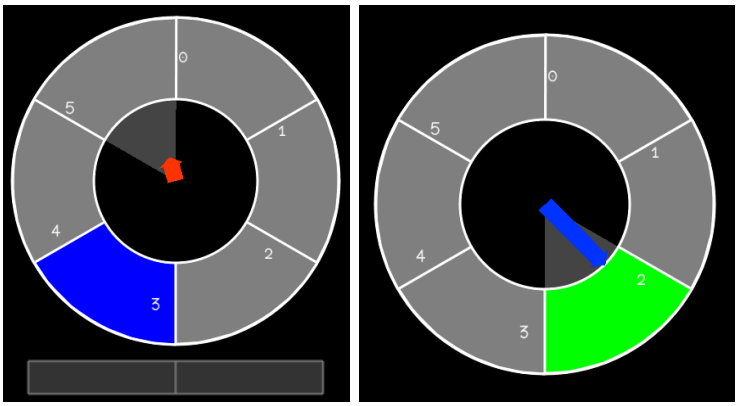


Figure 5.1: Screen shots of the Rotate-Extend (REx) paradigm. Left: arrow in rotation mode with the target segment highlighted in blue with the integration feedback bar on display. Right: arrow in extension mode, selecting the correct target segment which turns green.

preference; in cases where there was a bias in the classifier (where the participant found it easier to reach the threshold for the class), this class was used for rotation. The arrow was set to turn 2.5 seconds per segment, while the extension was set at 0.3 seconds. The integration of the arrow used the exponential smoothing function described in Chapter 4, given by

$$Y_t = \alpha Y_{t-1} + (1 - \alpha) X_t$$

where Y_t is the integrated classifier output at time t, X_t is the raw classifier output at time t, and α is the rate of integration between 0 and 1.0. High values of alpha are smoother but require a longer time to attain a threshold, while low values are more sensitive to the classifier outputs at each time step. At the start of a trial, the integration value was reset to 0.5. The target was displayed to the user, and after 1 second the arrow began to behave according to the value of the classifier output. If the integrated classifier value was under the rotate threshold, the arrow rotated, while if it reached the extend threshold, the arrow extended. If the value was in between the thresholds, the arrow would stop moving. The initial values were 0.4 for the rotate threshold, and 0.8 for the extend threshold. Table 5.1 summarises the parameters that could be tuned for the REx paradigm.

5.3 Method

5.3.1 Participants

10 healthy participants (mean age 30.8 ±11.45 SD) who had previously been trained to perform MI with the calibration tasks participated in the experiment. All participants had previously had at least one session during which they were exposed to the use of the REx paradigm. For the session used to generate simulation data, 30 calibration trials were first carried out, followed by performing test runs of the REx trials, where the parameters for

Table 5.1: Initial parameters set for the Rotate-Extend (REx) paradigm.

Parameter	Description	Initial value
MI classes	Mental class for rotate or extend	N/A
Integration rate	α parameter in the exponential smoothing function	0.95
Rotation threshold	Value of classifier output under which the arrow rotates	0.4
Rotation time	Minimum time taken to rotate through each segment	2.5 sec
Extension threshold	Value of classifier output above which the arrow extends	0.8
Extension time	Minimum time taken to extend the arrow to reach the inner circle	0.3 sec

the rotation and extension speed and thresholds were configured individually to the user. Participants carried out 48 REx calibration trials for 6 segments, with breaks in between every 12 trials.

5.3.2 Models and parameters.

The simulator models from the previous chapter were used in offline simulations to predict the task performance of individual users in the REx paradigm. The simulated classifier outputs from the models can either be directly fed into the paradigm to simulate the system in offline mode, or used to generate statistics that are used to drive an abstracted model of the paradigm (e.g. a finite state machine abstraction). Both techniques were used in the simulation experiments. Some statistics can also be abstracted from the real data. Several model parameters relating to the user's control of the paradigm can be identified:

- **Time taken to reach rotation threshold from idle state.** This determines how easily the participant can put the arrow into rotation mode when the trial begins. If the first segment is not the target, the participant is likely to make an incorrect selection if they cannot achieve this in time. For the direct simulation method, this does not need to be estimated. For an abstracted model, an appropriate distribution can be generated from the continuous simulator or from real data.

- **Time taken to reach extension threshold from idle state.** If the target segment is the first in the wheel, this determines how easily the participant can select the first target. Again, this can be generated from either the continuous simulator or from real data.

- **Time taken to reach extension threshold from rotate state.** This partly determines whether the user can select the intended segment with the rotation rate

of the arrow within the time given for the segment. The delay occurs because of the time needed to switch mental states and the time needed for integrating the classifier output. Again, this does not need to be estimated for the direct simulation method. One could also model the accuracy of switching; however, the delay time can also incorporate this inherently if one assumes that eventually a person would be able to make the switch from rotation to extension given an infinite length of time. In this case an abstracted model of the time distributions can be used; a sample drawn from this distribution that is longer than the rotation time allocated per segment would be counted as a miss.

- **Point of time at which the user consciously switches mental states.** As there is some delay in the time required to switch between the rotation mental state to the extension mental state, the point in time which the user switches again affects whether the intended segment can be selected. The user may have to compensate for the delay by intentionally switching before the arrow has reached the target segment. This must be estimated or obtained from the user as there is no way to pinpoint exactly when the person started to switch mental states.

- **Ability to switch back to rotate state.** The correct target may be missed if the time taken for the extension exceeds the rotation rate. This may be due either to the user being unable to produce the correct mental state for extending the arrow; alternatively, it may just take them a longer time to switch than the rotation rate allows for a particular target segment. In the latter case, if the user does not switch back to the mental state required to rotate the arrow again, the arrow may extend and hit the segment after the target. This can either be modelled with a probability indicating how likely the switch is to take place, or a distribution of times when the user will switch states. The values must be estimated as there is no way to obtain this information prior to the experiment. This is because in the binary calibration trials, there is no need for the user to switch mental states.

- **Number of false extensions.** The expected time to reach a false extension in the rotate class determines how often there will be false selection while the user is rotating the arrow round the wheel. This can be extracted from the data either through direct simulation or by abstracting from the continuous simulator.

The times taken to reach an extension or rotation threshold can be directly obtained from data collected using the binary calibration trials. One caveat is that the data from the calibration trials may not be long enough to obtain enough information about the longer, continuous imagination of a particular mental state, which is in particular needed for the mental state used to rotate the the arrow. The point of time that the user consciously decides to switch cannot be estimated from the calibration trials, as it depends on how the user perceives when they need to switch, or their mental model of how best to achieve the extension; and the ability to switch back to the rotate state if the correct target segment is

missed cannot be extracted from the binary calibration task. Where used as an input to the model, the expected time to make a false extension is thus extracted from the continuous simulators.

As this was a first attempt to determine the efficacy of using data from individual participants in a binary calibration task to predict their performance in a novel selection mechanism, several models were developed and used to generate simulation data for comparison. The intention was to provide recommendations for further research in developing simulation models. The *Markov Prob* and the *IAAFT* models described in Chapter 4 (Sections 4.5, 4.6.1) were used as direct input into the offline simulation of the paradigm ('continuous' models). Since this method may be time-consuming and it was not clear that this would generate the best results, models which abstracted user statistics were also explored, which are simpler and thus allow for a faster simulation run-time. (On the other hand, it may take some overhead to generate the statistics from the simulator models.) The abstracted parameters were fed into a Finite State Machine (Figure 5.2, 'abstracted' models). Finally, ensemble modelling was also used in order to find out if combining the results of different models would improve the predictions over individual models.

Three methods were used to generate simulated data for comparison with real data. The first method ('*Continuous simulator*') used the classifier output simulators from Chapter 4 as direct input into the REx paradigm. The second method ('*Abstracted simulator*') used abstracted data from the simulators, with time distributions for rotation to extension times, false extension times and time to extend if a segment was missed. The third method ('*Abstracted real*' and '*Abstracted min simulator*') simply used the distribution of rotation to extension times, ignoring any false extensions that might occur. If the sample drawn for a time-to-extend was longer than the rotation time, the model switched back to the 'rotation mental state' with a fixed probability of 0.7. The '*Abstracted real*' model abstracted summary statistics from the real binary calibration data, while the '*Abstracted min simulator*' models used summary statistics abstracted from the simulators. Since the summary statistics used were time distributions, each parameter was modelled using a Gamma distribution, where the parameters were estimated using the Metropolis-Hastings algorithm. The final parameters α and β were the mean parameters sampled from the 160th to the 200th sample, with a step size of 5 samples, for three runs of the algorithm (the mean of 120 samples in total). Where the distributions were generated from the continuous simulators, 500 samples for each parameter were first generated and used to estimate the parameters for the Gamma distribution. As 2 simulator models were used, a total of 7 models were compared. For each model, 800 trials were simulated for each segment at switching times of 0, 0.5, 1 and 1.5 seconds before reaching the target segment, giving a total of 100 runs of 8 trials per segment. This was compared to 2 runs of 24 trials (giving a total of 48 trials, 8 trials per segment) for the real data.

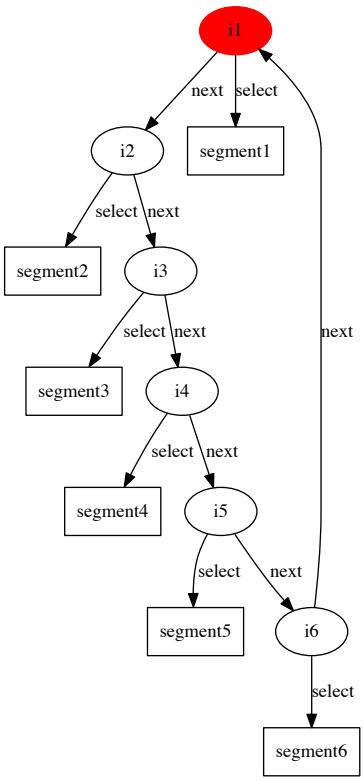


Figure 5.2: Finite state machine used for simulating the REx selection mechanism using the abstracted models. The states $\{i_1,...i_6\}$ represent the wheel segment that the arrow is currently pointing at. At the start of a trial, the arrow is positioned at i_1. For a given state i_s, continuing to rotate the arrow for the duration of the rotation rate triggers a 'next' transition, which leads to the next state in the FSM i_{s+1}, while an extension triggers a 'select' transition which terminates the trial and returns the selected state, $segment_s$.

5.4 Results

5.4.1 Comparison of overall accuracy.

Table 5.2 is a summary of the comparison of the mean average error (MAE), error range and the number of participants for whom the actual selection accuracy was contained within the 95% percentile prediction interval (PI) of the simulated data. The predictions from the individual models taken on their own are shown in Figure 5.3. The individual models with the best predictions are the abstracted min Markov and the abstracted real: both models have the lowest MAE of 13% and 11% respectively, and the prediction intervals are wide enough to capture the overall selection accuracy for 8 out of the 10 participants. The performance of the other models in terms of comparison with the actual data are comparable. Arguably, the worst individual models are the continuous IAAFT and the abstracted IAAFT.The continuous IAAFT has the highest MAE of 22% and consistently predicts a selection accuracy lower than the true accuracy, with the exception of one participant (p008). The abstracted IAAFT model was able to predict the scores for 3 participants with a MAE of 18%.

An ensemble model (averaging the scores for several models together) improves both the MAE and the number of participants' scores that are predicted. Figure 5.4 shows that overall, combinations of the IAAFT and the Markov models are comparable, with the Markov models generally making more optimistic predictions than the IAAFT models. This is generally due to the conservative estimates of the continuous IAAFT model. Figure 5.5 compares the actual data with three combinations of models: combining all 5 abstracted models, all the simulator models and finally all the models. Overall, the two best models are the one which amalgamates scores from all the simulator models, and one that uses all 7 models. Both results are very similar, with all 10 participants' selection accuracy contained within the PI and an average MAE of 10%. The abstracted models consistently overestimate the selection accuracy of participants with the exception of pAR; however again with the exception of p008 all the participants' scores are well bounded within the models' predictions.

Since the simulation runs for the continuous models took longer than for the abstracted models, trials which took longer than 240 seconds were terminated and considered to be wrong. No such restriction was placed on the trials for the abstracted models. The discrepancy between the overall accuracy in the continuous and abstracted simulator models can be explained for 3 participants: in the continuous IAAFT and the abstracted IAAFT models for p004 (7.0% of trials scored wrong due to timeout), and between the continuous Markov and the abstracted Markov models for pDT (11.9%) and p006 (36.3%). The discrepancy between the continuous and abstracted models for other participants seems, for the most part, to be due to a larger proportion of trials selecting the segment before the target. This occurs because for the abstracted models, the implementation of the simulation algorithm simply selected the target segment if the time-to-extension was less than the switching time, rather than selecting the previous segment.

A couple of trends can be found with regard to individual differences in the prediction of overall selection accuracy. The predictions for participants p008 and p012 are consistently overestimated, while that for pAR is almost consistently underestimated. Particularly for p008 and p012, this suggests that an additional model may be required to predict the scores for these participants. If their results are removed, the MAE is reduced to within 7% (range -8–10%) for all the abstracted models, 5% (-17–4%) for all the simulator models and 4% (-16–6%) for the combination of all models. Most of the combinations of models also predict the selection accuracy well with a MAE within 9%; however here the accumulated Markov models perform slightly better (MAE 7%, range -10–16%) than the accumulated IAAFT models (MAE 9%, range -24–5%).

5.4.2 Comparison of accuracy for each target.

To provide a more detailed analysis of how the overall selection accuracy is achieved by the actual participant and simulated models, the accuracy for each target was compared. For each target segment, the KL-divergence between the actual and average simulated distributions of *segments selected* was computed. Table 5.3 compares the distributions for 4 representative participants. Several behaviours can be observed if a 'good' score is taken to be a KL-divergence for $\leq$0.5 for at least three of the six segments. Firstly, a single model might account very well for the observed selections for all targets (pAR, with the abstracted min IAAFT model). Secondly, none of the models may predict the user's behaviour very well (p012, p008). Finally, different models may account best for different targets. This is the typical behaviour observed in the majority (7 out of 10) of the participants. The best models may be spread across all models including the continuous models (p005, p009), or across the abstracted models (p001, pDT, p004, pAC, p006).

Thus the model that best matches selection accuracy and accounts for the underlying reason behind the performance for each target may be different. For example, for p005 the abstracted min Markov model matches the actual data with a KL-divergence of 0.02 for target segment 3 (a very close match), and 0.51 with the continuous Markov model. However, for target segment 4, the KL-divergence for the Abstracted Min Markov model is 7.32 while the continuous Markov model makes the best prediction at a value of 0.22. Similarly, for participant pDT (Figure 5.6), the abstracted Markov model best predicts the performance for target 2, the abstracted min Markov for targets 3 and 6, whereas target 4 is best accounted for by the abstracted IAAFT model (KL-divergence=0.17) while the other models do rather badly for that particular target (KL-divergence$\geq$0.73). It can be seen that the abstracted IAAFT model better captures false selections due to random extension of the arrow. This leads to the observation of a decreasing trend in accuracy as the target angle increases. The results provide preliminary evidence to suggest that for these participants, some cognitive factors influence a participant's behaviour in different targets.

As previously mentioned, the selection accuracy of p008 and p012 is consistently overestimated by the models by a large error margin. Although the overall selection accuracy is

Table 5.2: Comparison of mean absolute error (MAE), predicted range, and number of actual accuracies contained within the prediction interval (PI), for different models and combinations of models. Each individual model uses 100 runs of 8 trials per target segment, where the simulated participant switches mental state 0, 0.5 and 1.0 seconds before reaching the target segment (total of 300 runs). The real data consists of 48 trials (8 trials per target segment). The values are averaged over 10 participants.

Model(s)	MAE	Error Range	Num participants contained in PI (out of 10)
continuous IAAFT	0.22	(-0.50, 0.13)	6
abstracted IAAFT	0.18	(-0.24, 0.36)	3
abstracted min IAAFT	0.17	(0.00, 0.44)	4
continuous Markov	0.18	(-0.27, 0.43)	6
abstracted Markov	0.17	(-0.05, 0.43)	4
abstracted min Markov	0.13	(-0.04, 0.36)	8
abstracted real	0.11	(-0.09, 0.45)	8
continuous, abstracted IAAFT	0.17	(-0.37, 0.25)	6
continuous, abstracted min IAAFT	0.11	(-0.25, 0.29)	10
abstracted, abstracted min IAAFT	0.13	(-0.12, 0.40)	8
all IAAFT	0.13	(-0.24, 0.31)	10
continuous, abstracted Markov	0.15	(-0.16, 0.43)	9
continuous, abstracted min Markov	0.13	(-0.14, 0.39)	9
abstracted, abstracted min Markov	0.14	(-0.03, 0.39)	7
all Markov	0.13	(-0.10, 0.41)	8
all abstracted	0.13	(-0.08, 0.41)	9
all simulator	0.10	(-0.17, 0.36)	10
all	0.10	(-0.16, 0.37)	10

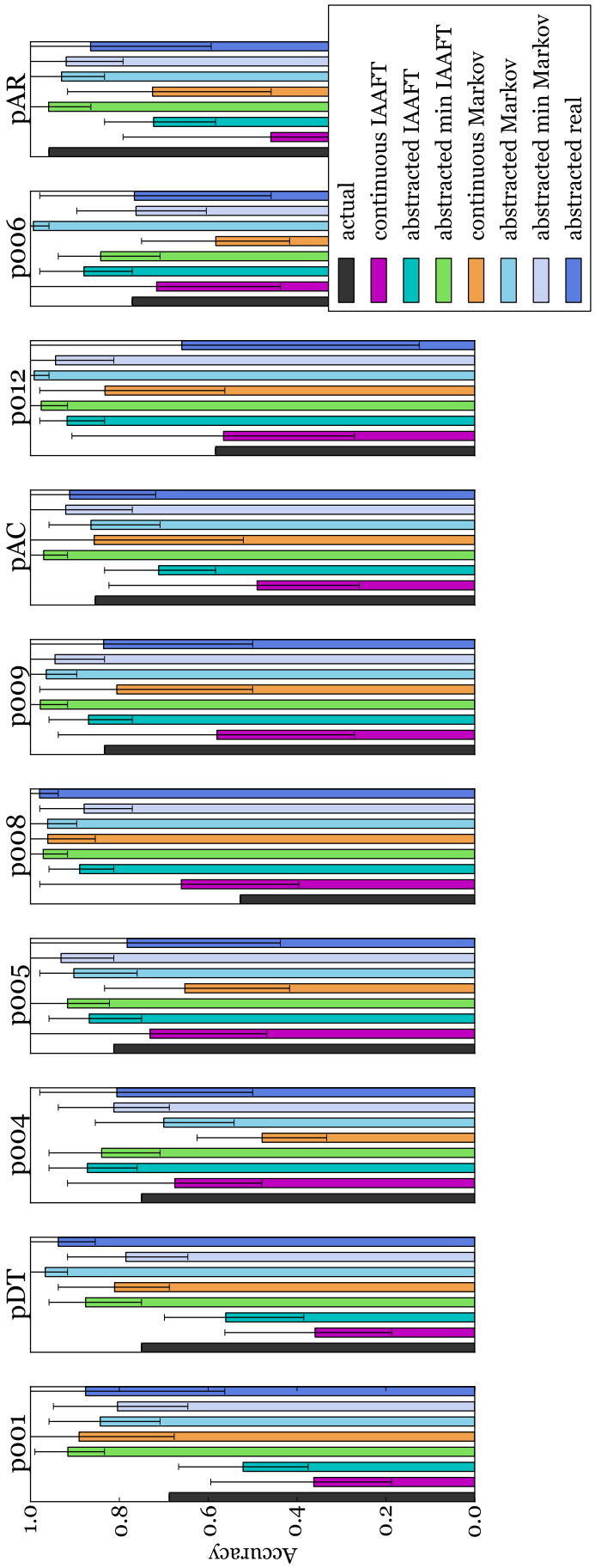


Figure 5.3: Comparison of the selection accuracy of the REx paradigm for actual and simulated data for each individual model, for each participant. The error bars for the simulated data show the 95% prediction (percentile) interval.

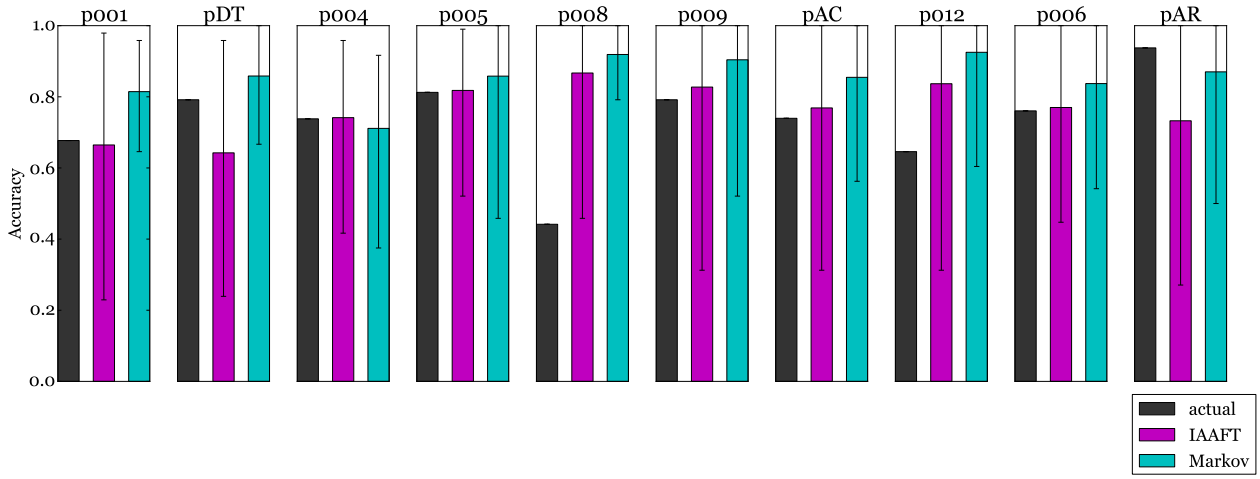


Figure 5.4: Comparison of the selection accuracy of the REx paradigm for actual and simulated data for the IAAFT and Markov models, for each participant. The error bars for the simulated data show the 95% prediction interval.

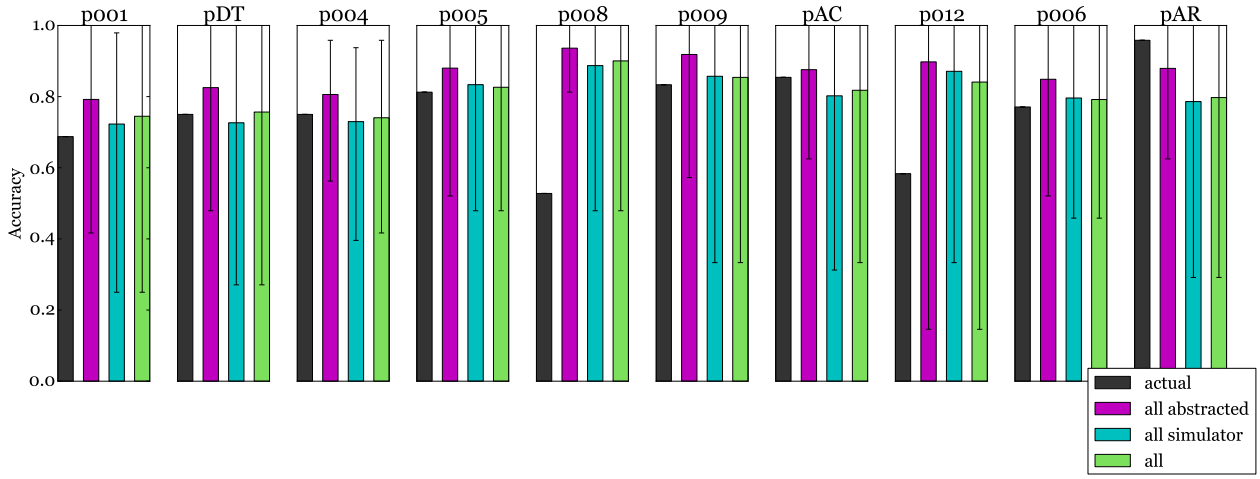


Figure 5.5: Comparison of the selection accuracy of the REx paradigm for actual and simulated data for the IAAFT and Markov models, for each participant, showing combinations of models with actual data: all abstracted models, all models using the simulators, and all the models.

predicted by the continuous IAAFT model, analysis of the KL-divergence of the selected segments for each target indicate that the model does not adequately account for the reason behind the decrease in expected accuracy for these two participants. In fact, the results are better modelled by replacing the rate of false extensions (which were estimated from the rotate state) with an idle state control. Figure 5.7 shows that this model consistently does better than the continuous IAAFT model. The average KL-divergence across all the targets for p008 is 1.45 using the continuous IAAFT model, and 0.64 using the abstracted IAAFT model for false extensions.

5.4.3 Comparison of time to selection.

This section considers the time taken for a participant to make a correct selection and how the data from the simulation models compare with real data. Table 5.4 compares the task timings compared with the real data across the different models. The geometric mean is used as as the summary statistic of central tendency as it is a useful approximation to the true median for small sample sizes (Sauro and Lewis, 2010). Comparison across models is computed by taking the median of the average geometric mean error over the simulated runs for each participant and each target. In terms of the median average geometric mean error, the best individual models are the continuous IAAFT (median deviation of 2.33 seconds), continuous Markov (2.98 seconds) and the *abstracted real* (3.91 seconds). Tabulating the percentage of actual geometric means contained in the 95% prediction interval (PI) for each target and participant for each model show that the best individual model predictions are again the *continuous IAAFT* (67% of targets contained in the prediction interval), *continuous Markov* (70%) and the *abstracted real* (70%) models. As indicated by a Box plot (Figure 5.8) showing the distribution of errors across the models, the continuous models generally overestimate the actual data while the abstracted models are more likely to underestimate the geometric mean on average.

A higher mean or median than the actual time is indicative of a longer tail than the actual data, while a lower mean or median may be indicative of either a shorter tail than the actual data, or an underestimation of the peaks in the selection time which represent when the participant is able to select the segment (taking into account the time it takes to loop around the wheel). There are few instances where the initial peak selection time is notably greater than the real data. A typical example is shown for participant p001 for target 4 (Figure 5.10). It can be seen here that the estimation of the peak timings are well captured by the continuous models, and that the abstracted models underestimate the loop time slightly. This is likely due to their not modelling the time during which the integrated classifier output is within the rotation and extension thresholds (i.e. when the arrow is not moving). Finally, the Markov models are seen to have longer tails than the IAAFT models. The benefits of such a long-tailed distribution are seen here where it has taken this participant around 70 seconds to select the correct target: the predictions from the Markov models contain this value while the other models do not.

Table 5.3: KL-divergence scores, comparing the distribution of segments selected for each target segment (for actual and simulated data across individual models). The best model for each target is asterisked. Data for 4 participants are shown, one for each category of observed behaviours (see Section 5.4.2 for details).

Participant, Target	continuous IAAFT	abstracted IAAFT	abstracted min IAAFT	continuous Markov	abstracted Markov	abstracted min Markov	abstracted real
pAR T1	0.06	0.06	**0.00***	0.01	0.02	0.01	0.02
pAR T2	0.87	0.11	0.05	0.58	0.04	0.02	**0.01***
pAR T3	0.91	0.29	**0.04***	0.38	0.08	0.11	0.18
pAR T4	1.03	0.42	**0.05***	0.41	0.08	0.10	0.17
pAR T5	1.15	0.49	**0.05***	0.40	0.09	0.11	0.17
pAR T6	0.93	0.48	0.05	0.23	0.08	0.01	**0.01***
p012 T1	1.93	**0.60***	1.93	1.93	1.93	1.93	1.96
p012 T2	**0.82***	1.43	6.17	2.56	5.01	6.07	6.06
p012 T3	0.73	0.94	4.17	**0.70***	4.30	4.08	4.14
p012 T4	**0.77***	1.09	6.16	1.47	2.25	6.07	3.41
p012 T5	**0.21***	0.46	4.44	0.63	4.34	4.55	4.51
p012 T6	**0.61***	1.12	8.43	1.94	2.72	8.32	3.60
p005 T1	0.01	0.05	0.01	0.02	0.03	**0.01***	0.02
p005 T2	**0.02***	0.82	4.12	0.07	0.84	4.11	4.25
p005 T3	0.38	0.14	0.11	0.51	0.10	**0.09***	0.31
p005 T4	0.75	0.05	**0.02***	0.55	0.07	0.02	0.06
p005 T5	0.70	1.64	6.95	**0.22***	1.57	7.32	7.12
p005 T6	**0.13***	0.28	2.02	0.36	0.26	2.00	2.19
DT T1	0.51	0.33	0.02	0.14	0.02	0.03	**0.01***
DT T2	1.02	0.51	0.15	0.22	**0.04***	0.29	0.07
DT T3	0.86	0.34	0.10	1.55	0.38	**0.07***	0.18
DT T4	0.52	**0.17***	0.91	2.17	2.90	0.73	1.33
DT T5	1.00	0.46	**0.08***	1.38	0.42	0.08	0.20
DT T6	0.64	0.37	0.41	1.79	1.08	**0.18***	0.65

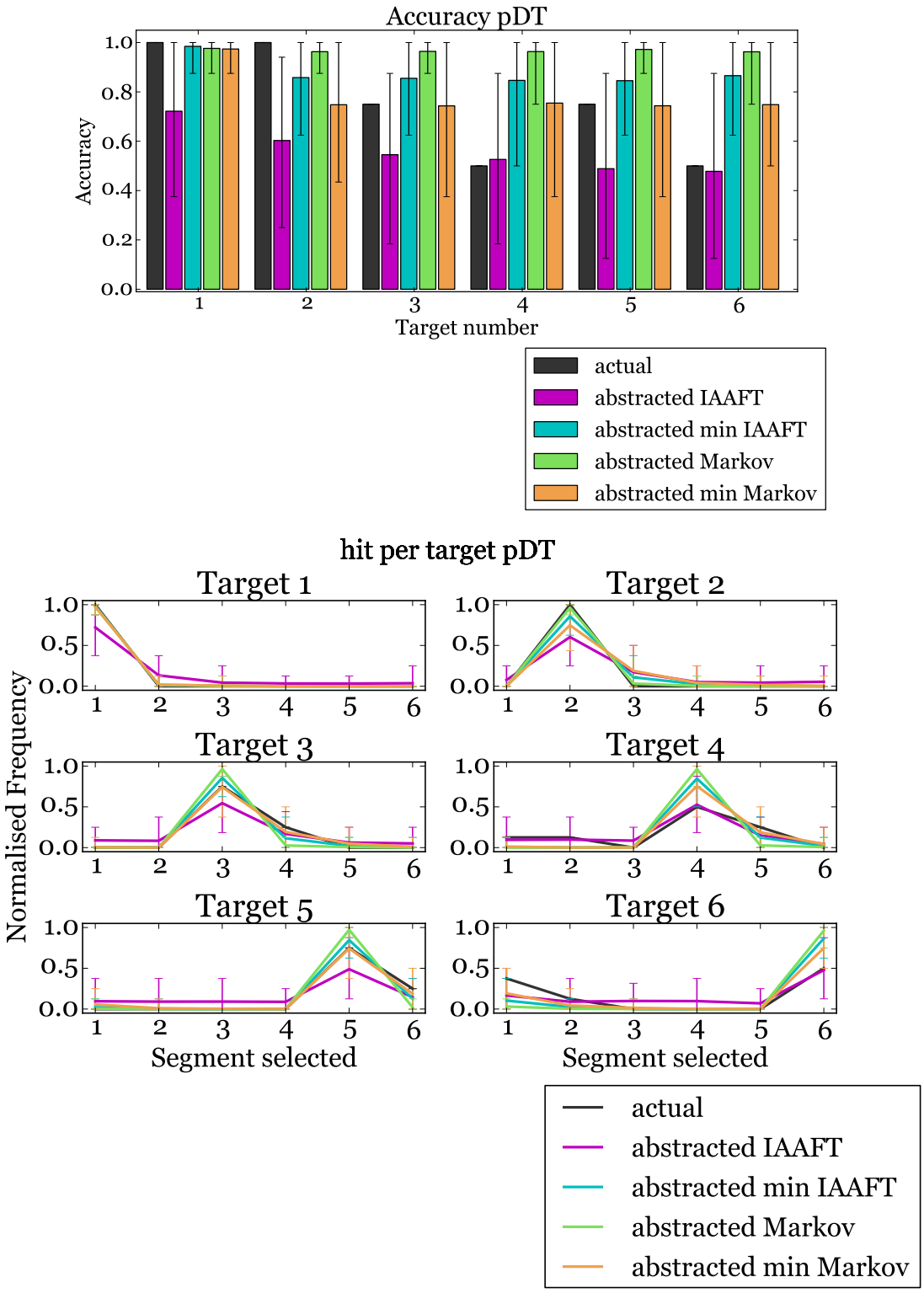


Figure 5.6: Target accuracy (top) and detail of segments selected for each target segment (bottom) for participant DT, for individual abstracted models. Different models account best for the selection of different target segments.

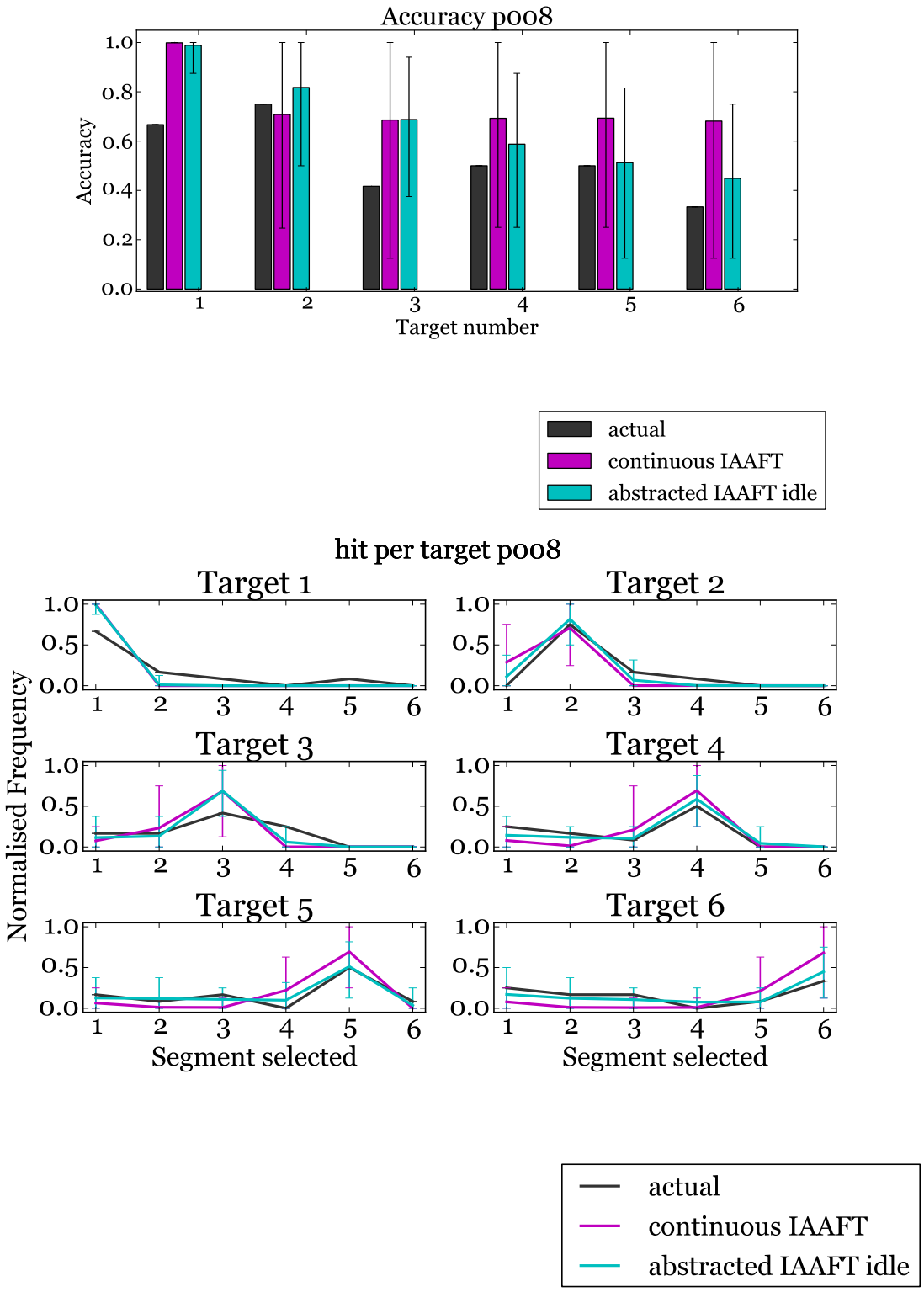


Figure 5.7: Continuous IAAFT model compared with abstracted IAAFT using the false extension rate during the idle (no control) mental class.

However, the tail of the distributions sometimes does not seem to match the real data. For p006, for example, the longest time taken to make a correct selection was 33 seconds for target 6, while the longest simulation prediction was over 700 seconds. In this case, the combination of the high selection accuracy and the much longer tail of the time-to-selection reflects that false selections are not predicted by the models. It would be useful to truncate the timings above a value at which point a person is likely to give up or the trial should stop to avoid fatiguing the user. This should be taken close to the longest time taken for a participant to make a selection, which was 121.63 seconds for participant p008. Comparisons across individual participants shows that the long tails in the model distributions occur mostly in p006, pDT and to a lesser extent p004. As the individual models are seen to have different characteristics, averaging the results of the models are likely to provide better estimates. Figure 5.9 shows that across all models for each participant, the actual geometric mean timings are generally within ± 5 seconds of the simulated predictions. Combining models also produces results that are closer to the true results such that the prediction intervals contain most of the actual geometric means for each of the IAAFT (97%) and Markov (90%) taken together, all of the actual geometric means for the models combining abstracted only, simulator only and all models.

5.4.4 Effect of switching time on accuracy

Although the previous models averaged across the values used for switching between mental states for convenience (0, 0.5 and 1.0 seconds before the target segment is reached), closer inspection of the simulated data with respect to the switching time does in some instances reflect the user's behaviour better for different targets. Figure 5.11 shows for p009, the IAAFT models separated by time at which the person switched mental states accounts for the different in selection accuracy for targets 4 and 5. Here, the closest match to the segments selected for target 4 is a switch time of 0 seconds: the KL-divergence for the 0-second model is 0.42, while that for the 1.0-second model is 0.71. Conversely, for target 5, the 0-second model has a KL-divergence of 0.46 while the 1.0-second model matches the actual data very closely at 0.10. (A very similar result is found for the Markov model, although the simulation results for this model do not show any differences between a switch time of 0 and 0.5s.) This may indicate that for target 5, the participant had more difficulty anticipating the correct time to begin extending the arrow.

5.5 Discussion

Comparisons between simulated and real data show that the individual models capture different possible participant behaviours. For most participants, the best individual model for predicting accuracy and task timing is the *abstracted real* model, which simply models the time taken to reach the extension threshold as estimated from the real data of binary calibration trials, and sets a probability of successfully switching back to the rotation state (if the correct target segment is missed) at 0.7. However, it does not generate some behaviours

Table 5.4: Summary of simulated timings compared with actual data.

Model	Median Geometric mean error	Median Absolute Geometric mean error	Quartile range	Percentage contained in PI
continuous IAAFT	1.79	2.33	(-1.44, 29.25)	0.67
abstracted IAAFT	-3.72	4.28	(-9.89, 2.33)	0.47
abstracted min IAAFT	-3.80	4.17	(-9.08, 2.84)	0.38
continuous Markov	2.17	2.98	(-1.71, 48.72)	0.70
abstracted Markov	-2.49	4.11	(-6.53, 21.28)	0.65
abstracted min Markov	-2.91	3.91	(-6.61, 13.40)	0.57
abstracted real	-3.12	3.12	(-9.67, -0.91)	0.70
all IAAFT	-2.38	4.89	(-7.26, 10.82)	0.97
all Markov	-1.44	2.91	(-4.18, 27.12)	0.90
all abstracted	-3.04	4.28	(-7.71, 7.79)	1.00
all simulator	-1.82	3.70	(-5.59, 17.45)	1.00
all	-1.92	3.33	(-6.11, 14.30)	1.00

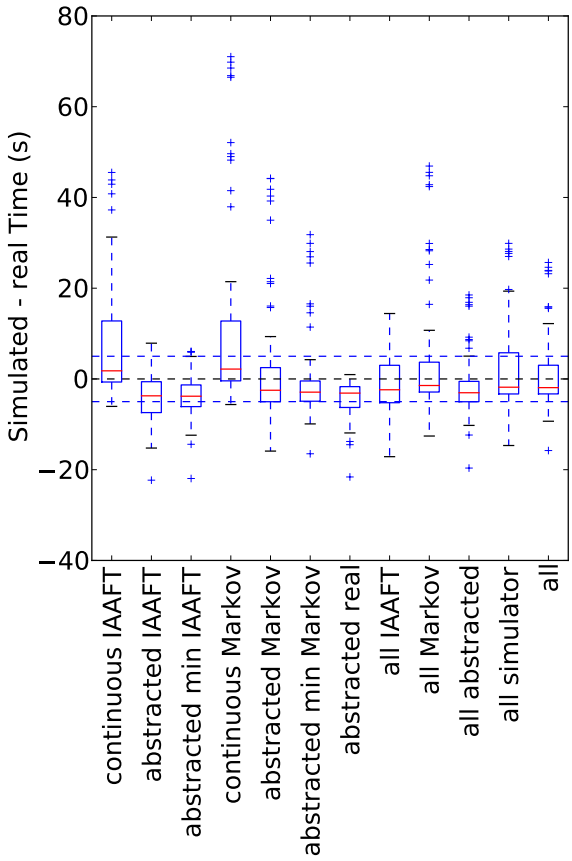


Figure 5.8: Average geometric mean errors of timing across models, for each target segment and participant.

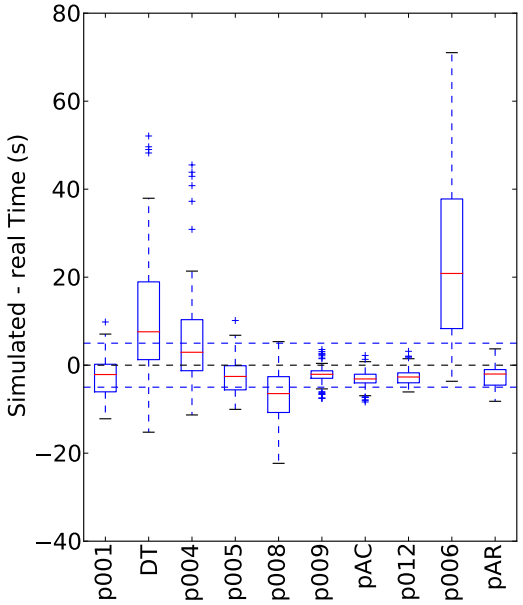


Figure 5.9: Average geometric mean errors of timing across participants.

that are observed in the actual participants' data, such as the rate of false extensions. The model that provides the most conservative estimates of task accuracy is the continuous IAAFT model, which has the tendency to underestimate task accuracy because it overestimates the rate of wrong selections due to extending the arrow too early. The continuous models tend to predict the peak distribution of task timings more accurately than the abstracted models, while sometimes having flatter distributions. This leads to the continuous models tending to overestimate slightly, and the abstracted models tending to underestimate slightly, the predictions of the geometric mean across targets for each participant. Thus, the individual models capture different user behaviours and performances such that no single model successfully accounts for the performance of all participants. However, by averaging the simulation results over combinations of models, the overall performance of individual participants is better predicted, both in terms of task times and selection accuracy.

The combination of abstracted models generally overestimated the selection accuracy of participants (with the exception of pAR). This is expected as some phenomena were not modelled by the abstracted models: that the previous segment of the target could be falsely selected if the user switched mental states too early, and that the first segment of the wheel is sometimes falsely selected. Participants' comments in the experiments indicated that this second phenomenon was due to not being well prepared to make a selection, explaining why for a few participants the first segment of the wheel was most often mis-selected. On the other hand, setting the probability for switching back to the rotation state to 0.7 (for the *abstracted min simulator* and *abstracted real* models) proved to be a useful approximation for simulating the observation that the segment after the target was sometimes selected.

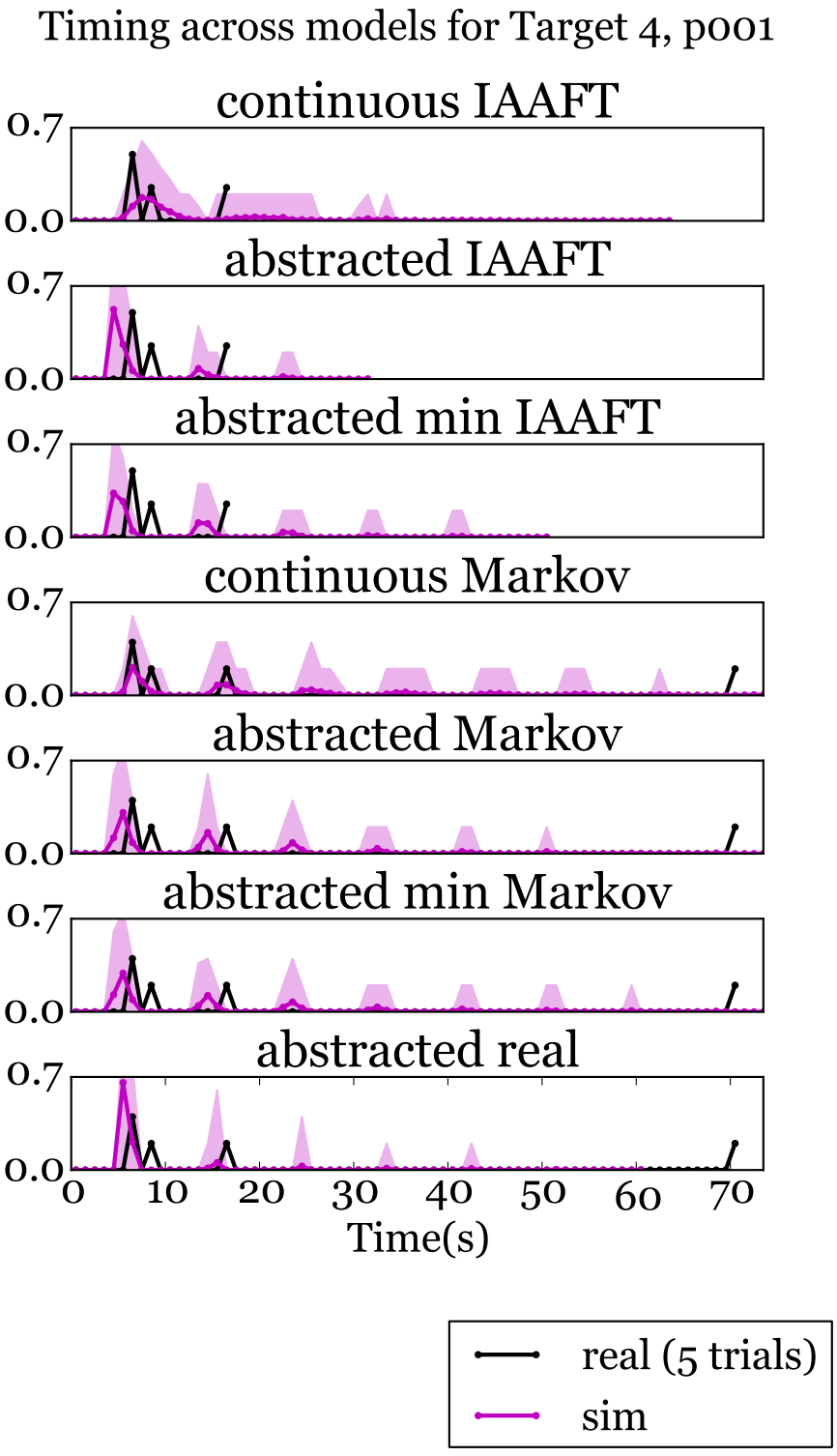


Figure 5.10: Comparison of models showing the typical distribution of time to make a correct selection for the simulation models. In general, the continuous models match the actual timing of the peaks in the real distributions while the abstracted models slightly underestimate the timing. The shaded regions represent the 95% PI of the simulated data.

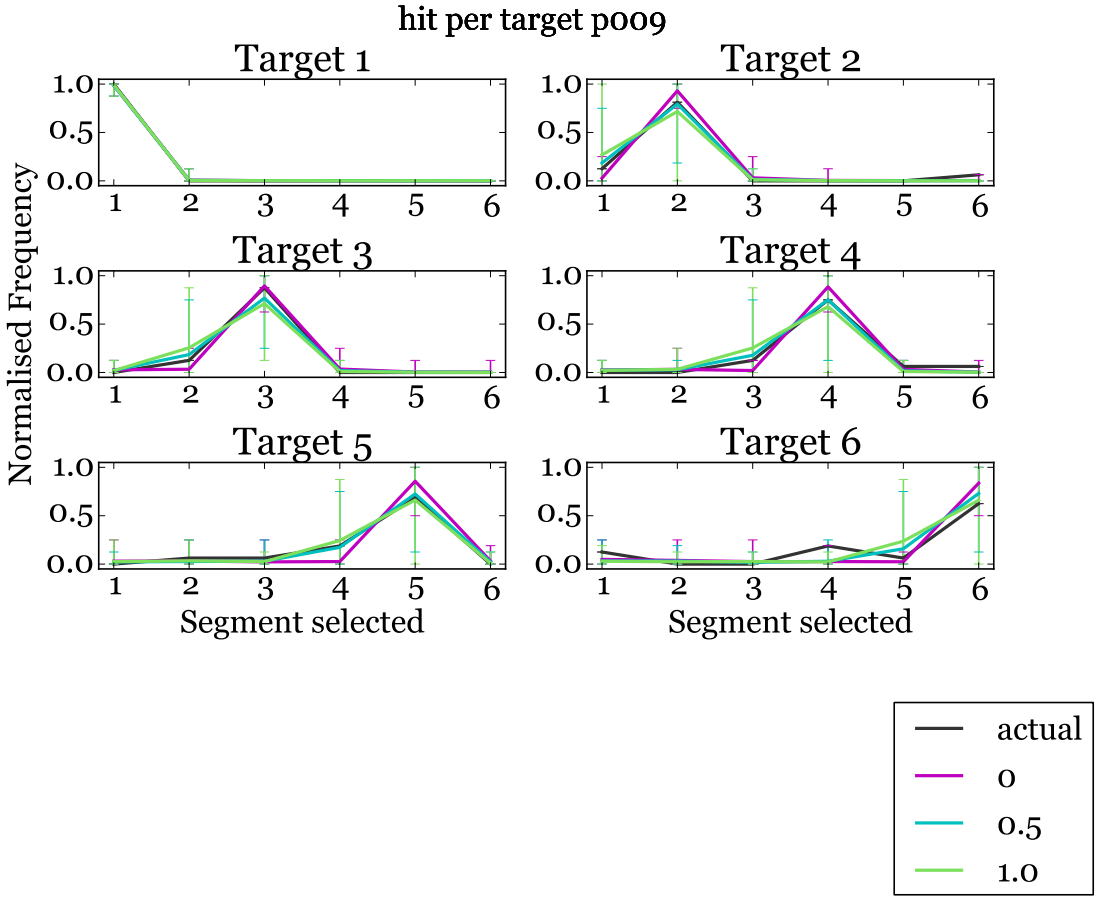


Figure 5.11: Effect of different switching times (0, 0.5 and 1.0 seconds before reaching the target segment) on selection of targets for p009. The figure highlights that a difference between when the participant switched mental states accounts for the difference between the segments selected for targets 4 and 5.

This occurs when participants have started to imagine the extend class, but due to the delay in switching mental states, miss the target segment; if they cannot then switch back to the rotation state in time, the arrow continues to extend and select the next segment. This phenomenon was also observed in the *abstracted simulator* models, where the ability to switch back to the rotate state was modelled explicitly with a distribution of expected times for each participant when the arrow would extend even if the participant switched back to the rotation state. The accuracy of these predictions in comparison to the actual data varied between participants and between models, indicating a scope for improving the models.

In general, the IAAFT models better captured the rate of false extensions occurring during the rotation time, leading to the observation of a decrease in selection accuracy as the target angle increased, while the Markov models were more optimistic in their predictions of selection accuracy. Depending on the participant, this influenced the models which provided the closest match to the selection accuracy. Interestingly, the MAE of the all the simulator models taken together is lower than either of the IAAFT or Markov models combined separately, highlighting again that the combination of results from different models strengthens the overall predictions. Combinations of models also better capture an overall performance as different models tend to account best for the selection accuracy of different target segments for each user. The fact that different models match the behaviour for different segments for each participant may indicate that the position of the target affects user perception. For example, for user p009, it is possible that the position of the target segment meant that her perception of timing was different, such that she started to rotate the wheel earlier for some targets, leading to a proportion of trials where the previous segment before the target was mis-selected.

The main limitation of attempting to simulate user behaviour for the REx paradigm from binary calibration trials is that information about the behaviour of the classifier output during a prolonged period of imagining the same mental state is not available. For some participants, this was nevertheless modelled very well by the simulator models. For a few participants, these deserve further consideration. For participants p008 and p012, it was seen that although the continuous IAAFT model is able to better account for the overall selection accuracy, the model falls short of following the trend of the actual data on inspection of the selection accuracy of individual targets. Thus, the two users may be thought to be in a category of users for whom a different model might be better used, namely a model which uses the false extension rate based on an idle (non-control) state. On the other hand, the task timings for p008 showed that there were instances of correct selections which took a long time, lasting as long as 2 minutes, suggesting that the issue was not that he could not maintain the correct mental state but that he was unable to switch at the right time. This is also supported by the underestimation of mean geometric mean task times for this participant. For p006, predictions of mean geometric mean task times much longer than the actual recorded data (>20 seconds longer on average), combined with the higher selection accuracy where this occurs for the *continuous Markov* model (where wrong selections were

due to terminating the trials after 240 seconds) and the *abstracted min Markov* model, indicates that the false extension rate during rotation, the ability of the participant to switch back to the target, and the time taken to reach the extension thresholds, were not sufficiently represented with the models.

The combination of different models allow one to strengthen their predictions about performance. Unfortunately, in this case this also comes at the cost of the time required to perform simulations. The *abstracted min simulator* and *abstracted real* models, which use distributions of the time taken to switch between mental states but not the false extension rates, largely provide a good approximation to real data in terms of the selection accuracy and time-to-selection. Although the continuous models should by intuition be the best predictors of performance as they are intended to directly model the classifier output, they take longer to run and are not necessarily better predictors of performance. The continuous simulator requires simply generating classifier output and inputting these directly into the REx paradigm, while there are more overheads of abstracting the statistics and developing the finite state machine and behaviour. However, each run of the continuous simulator takes longer than the abstracted method and thus a larger number of simulations can be run with the abstracted models in a shorter time. In addition, by using the abstracted models, different aspects of control can be isolated in order to find out what the important elements are for controlling the paradigm.

The simulation exercise highlights several applications of offline simulation. In the first instance, simulation of a novel paradigm can be used to identify the various behaviours that might be expected. For example, the point of time at which the participant switched mental states had an effect on the selection accuracy, and simply setting a probability of being able to switch back to the rotation state upon missing the target segment accounted for selection errors. Thus, although offline analysis does not provide indications of workload, it can be used to establish an expected task performance for a given user with known control characteristics. This was supported by McFarland and Wolpaw (2003), who showed that the trends in the selection accuracy could be predicted by simple simulations. However, as the paper explained that individual differences give rise to the need for optimising the interface parameters, it would be useful to automatically infer what these might be for a particular participant. Although this was not addressed in this experiment, the simulation results closely match performance in the novel paradigm in which the parameters were individually tailored to each participant. These initial results are promising as they indicate that by refining the simulation models, one might be able to select the optimal parameters by simulating the task performance with various interface parameters.

Another use of simulation is that having fine tuned the models to find the the control characteristics that provide the best predictions of task performance for a given paradigm, these can be used to provide more accurate predictions of task performance for a particular user. For example, as evidence suggesting that task accuracy improves during a more motivating task as compared to calibration trials (Leeb et al. (2007); unpublished data from

REx trials), the abstracted models may be thought to be good indicators of performance where participants are performing well: increased motivation may lead to better control of the rotation, and thus fewer false selections would be made. Similarly, being able to choose the target rather than being shown which segment to select may enable the participant to choose the correct one. However, the results from the abstracted models indicates that there is a baseline level of expected errors that would be made by each participant. Future work could examine the relationship between the accuracy of simulated data and actual data for calibration trials compared with performance in a real application. Since individual variability between participants and sometimes within participants is large, having useful predictions of task performance can lead to improvements in the paradigms, or be useful for optimising parameters for the paradigm, as previously mentioned. Identifying categories of participants may also lead to better predictions about how well they are able to control different paradigms, as what works for one participant may not be best for another. Initial parameter values or paradigms can be selected based on identifying classes of users, which could then be fine-tuned for each participant.

Since a small number of actual trials were performed by each participant, the results from the simulation data should be taken with caution. Typically in the experiments reported in the BCI literature, individual participant data is accumulated over several sessions of trials. Here we only used data from one session; nevertheless the simulation results are promising as for most participants the selection accuracy could be well predicted. This provides a motivation for improving the models in order to make better predictions about performance. Calibration tasks can be better chosen that will provide the data required for the task estimation. It is possible that collecting data via a game, for example, rather than using the more monotonous calibration trials might be useful for predicting performance in other tasks. This data could be used to generate a user model that could be used to predict which system would be easiest for a particular user to use at a given point of time. The current simulation results provide a grounding which indicates that it would be valuable to use these trials to optimise the performance of other interfaces. In addition, the rather more motivating trials may provide more useful data as they might match actual performance in an actual application which would be motivating to an end user.

5.6 Addendum: use of the online simulator for development.

The simulator described in Chapter 4 was used extensively in an online manner in the development and debugging of the REx paradigm and its experiment, and the music player described in the next chapter. In the first instance, this led to the conclusion that the REx paradigm could be controlled using the BCI system. Some observations that were made during use of the online simulator were used to make changes to the system before asking people to use it in a real experiment. These included usability issues such as the timing of when trials should start and stop, and other minor bugs such as visual disturbances. Parameters used for the selection mechanism were also experimented with in order to determine suitable initial values. All of this was possible because the classifier output values

were fed directly into the system such that the possible effects of delays and fluctuations in the signals could be experienced without having to use a BCI.

Three participants used the simulator before trying out real BCI trials and provided comments on the similarities and differences they felt between the two. An interesting phenomenon that had been uncovered during development and debugging was that when switching between left and right classes in the REx paradigm, it was easiest to 'relax' to an idle state by releasing both keyboard keys before depressing the right shift key for the right (extend) class. When participants using the simulator had trouble with extending the arrow, this strategy was recommended as the useful thing to do. Although it is difficult to say how generally this strategy applies, one person who participated in BCI trials after using the REx paradigm remarked, 'your advice about having to let your mind rest/go blank before extending totally applies here as well. I think it's even more important when using BCI.' On the other hand, he commented that it was easier to switch between mental states than in the simulator.

Other potential differences between the simulator and the real BCI could be identified. One participant mentioned that it the REx mechanism was slightly more frustrating in real BCI than with the simulated BCI, while another participant clearly enjoyed the BCI trials more than using the simulator as he expressed, 'I can't get over the fact that I'm sitting here just thinking about moving the cursor and it's actually moving'. Thus, the novelty of actually using one's brain to control the system cannot be simulated by a simulator. On the other hand, it is difficult to identify whether differences in the perception of control arose from weaknesses in the simulation model or individual differences. Since the simulator was intended to represent one particular user, it is possible that any given user could perform better or worse. Validation of the simulator is therefore necessary if it is to be used to capture subtle differences in individual performance. Possible ways of validating the simulator for the purposes of the subjective feel of control have been described in Chapter 4.

5.7 Conclusions

This chapter attempted to use the data from binary calibration trials to predict the task performance (selection accuracy and time-to-selection) of a novel selection mechanism, the Rotate-Extend (REx). It was shown that individual models are collectively able to capture a range of observed behaviours, and that averaging over the predictions of the individual models (ensemble modelling) provides a closer match to users' actual performance than do the individual models on their own.

The data from binary calibration tasks were insufficient to adequately predict the task performance in the REx mechanism for a couple of participants. Some possible reasons are that the binary calibration tasks did not capture the user's ability to sustain imagination of the MI task which is a big part of controlling the REx selection mechanism, and that there are non-stationarities in the EEG which may affect task performance.

The chapter also highlights two potential uses of offline simulation. Firstly, offline predictions can be used prior to any real user trials to analyse and predict possible user behaviours and task performance for the design of interfaces. Secondly, models that provide the best predictions for actual user behaviour can be used to identify and select the most suitable paradigm to be used by a particular participant, even for a particular day.

Finally, a brief reflection on how the simulator was used in an online manner for debugging and development was provided. Several usability issues and bugs could be identified before real BCI trials were carried out, and participants' experiences of real BCI after using the simulator indicated that further validation of the simulator and investigation into the similarities and differences in user experience of control for actual and simulated BCI is required.

6 Applications II: Case Study in Developing a Music Player

Summary. This chapter demonstrates a use of the simulator described in Chapter 4, and the design process described in Chapter 2, to develop a BCI music player for end users with LiS. Video prototypes were shown to participants with physical disabilities in order to elicit feedback and comments about what they thought of the system (which would feed into the design of the system), and the online simulator was used in the lab and as a longitudinal study with healthy users in their own homes or places of work. The mixed-users, mixed-methods approach allowed us to form firm conclusions about the importance of control and individual differences in developing a BCI music player application for end users.

6.1 Introduction

As mentioned in previous chapters, the ability to communicate is of paramount importance in the quality of life for persons with LiS (Kübler et al., 2001). To this end, the vast majority of BCI systems have focused on developing spelling programs. However, a similar shift from developing 'useful' applications to interactive, social, hedonic systems which aim to provide users with other means to express themselves or communicate as seen in mainstream HCI has been shown in the BCI literature. For any given individual who finds themselves in a locked-in state, the functionality most desired is likely to be different. For example, the most successful BCI application shown to enhance the quality of life of a LiS person so far is the P300 brain-painting application (Münßinger et al., 2010). Other applications which can be considered to have an aim of providing entertainment and social inclusion include telepresence robots (Tonin et al., 2011) and games (Nijholt et al., 2009). Along these lines, music can be a large part of any given human being's life, but as far as is known, there has not been an attempt to design and develop a music player application for end users of BCI. It is not unreasonable to speculate that a person for whom music is a big part of their life, and who finds themselves in a locked-in state, would highly appreciate the ability to engage with music by selecting music to play.

Several issues emerge for developing a BCI application. For a 2-class MI BCI, some issues are that it is difficult to have an intentional non-control state, and that making selections is slow. Thus, the prototyping exercise aimed to determine the features of a music player that would be sufficient for use. In particular, the aims were to discover how end users

might choose to select music to play, how they might choose to cope with or use a system that randomly started and stopped music, and what functions or features found in typical desktop music players could reasonably be eliminated from a minimalistic music player.

One way to reduce the number of selections required to choose music might be to incorporate artificial intelligence techniques which generate playlists given a small number of inputs. For example, Tzanetakis et al. (2009) implemented a system based on self-organising maps to organise and retrieve music. The system can receive several types of input and was evaluated with one user with cerebral palsy who uses his lips to activate a switch as input into a computer. Apart from this example, there is a dearth of available music players which have been designed with disabled end users in mind. The current system sought to understand the functionality that would be important to end users of a music player. In particular, the functionality of the Moodagent[1] system is employed in the prototype. This is a commercial software that can classify all the music tracks in a user's collection with a range of features, including predictions of subjective 'mood', genre and tempo of the music.

Following a UCD approach, the design process proposed in Chapter 2 was used to obtain valuable input from users at different stages of design and development. Firstly, an initial user requirements capture questionnaire was used to explore disabled participants' current music listening habits. Video prototypes were also used to explore participants' reactions to possible ways in which a BCI-controlled music player might work. The results were compared with healthy and disabled participants' usage of a music player prototype. Healthy participants controlled the applications using various inputs including mouse, the MI-BCI Simulator described in Chapter 4 and real BCI, and disabled participants' used various control inputs as well as real BCI.

The number of participants involved in each phase of the process are summarised in Table 6.1 and Figure 6.1, which show the level of physical disability the participants had according to categories defined within the TOBI project (see General Discussion, Section 6.6, for discussion). For the purposes of the study, disabled participants were grouped according to the level of aid required to use a computer; for example, the largest group of participants had a Moderate disability where they were able to move around in a manual or electric wheelchair and use their arms but had limited or no grasp function. Note that a few participants took part in more than one study.

[1] www.moodagent.com

Table 6.1: Summary table of participants to all the user studies reported in this chapter, grouped according to the degree of disability with regard to the communication aid needed or assistive technology required to use a computer.

Degree of Impairment	Definition	Number of participants				
		Questionnaire	Video prototypes	Prototype (non-BCI)	BCI	Total
None	No known physical disability	0	0	10	6	15
Minor	Slightly impaired limb movement. Able to use verbal language	3	1	0	0	3
Moderate	Severely impaired lower limbs or no verbal language or severely impaired upper limbs or hemiplegic	2	3	3	1	6
Severe	Almost tetraplegic. Able to use verbal language and able to handle special input device, e.g. switch or joystick (two channels to control ATs)	6	1	1	2	9
Major	Tetraplegic with very little residual control of muscles and able to use verbal language or almost tetraplegic (able to handle special input device, e.g. switch or joystick) but no verbal language (effectively one channel to control ATs)	2	1	0	1	3
Locked in	Completely tetraplegic, no verbal language, very little residual control over a few muscles e.g. eye movement (only passive communication possible)	1	1	0	0	1

6.2 Initial Requirements Capture

In order to find out what level of access disabled users currently have to music and to determine what the goal or purpose of a music player should be, an initial requirements capture phase was carried out using questionnaires and interviews.

6.2.1 Methodology

A questionnaire was developed in conjunction with AT professionals, clinicians and researchers in the field, and administered to participants having some level of physical disability in the form of interviews. For users who are unable to speak and otherwise also have limited muscular control, a well-designed questionnaire can help to obtain information in the quickest and least effortful way. In such cases, the interviewer reads the question to the participant, then reads through the options linearly, pausing for the participant to indicate a 'yes' or 'no' to select or reject the option. 15 participants (5 female) aged between 18 and 55 (mean 40.9) answered questions about their requirements for a music player, however one set of results was removed from the data as that particular participant had no interest at all in using a music player. The reported results are thus results from 14 end users with varying levels of physical disability. The questionnaires were administered either by BCI researchers or AT professionals in Italy and Germany.

6.2.2 Results

Current access to music. Among the participants interviewed, 2 users had no access to music, 2 only had access to music through a stereo or CD player, while the rest had access to computer software either through standard keyboard and Mouse inputs or through assistive input devices such as joysticks, single switch devices or speech recognition software. All of the users who had access to music via a computer did so through standard software such as Windows Media Player, Winamp or iTunes, or websites such as last.fm and youtube.com.

When asked what they liked about their current access to music, users who had access to music through a computer indicated attributes such as 'fast, easy to use, reliable, and not dependent on a specific operation system', the ability to make a playlist with Youtube, and being able to organise music into folders that are easily accessible. Negative points about one user's current access to music included it being 'too exhausting and it needs verbal language recognition'.

Desired uses of a music player. From a list of six possible uses of a music player, participants were asked to choose their top, second and third reasons for using a music player. For each participant, the top reason was awarded three points, the second two points and the third one point. According to this scoring, the most desired uses of the music player are to have control over the music listened to and to pass the time enjoyably while alone (Table 6.2). Thus, listening to music through a music player is seen first as an individual

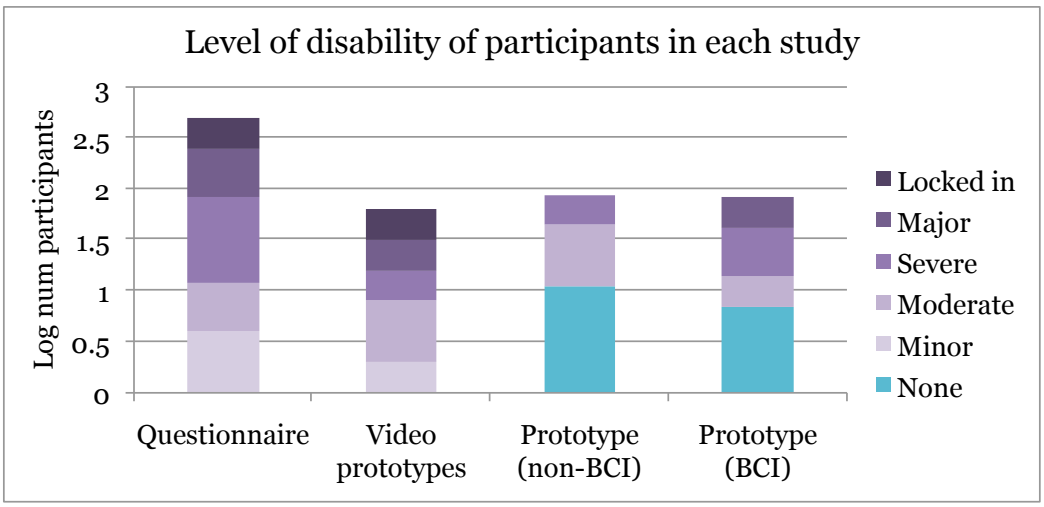


Figure 6.1: Log number of participants for each study, according to level of physical ability. Bars from left to right represent the stages of the design process from requirements capture, to prototyping, to evaluation of the final prototype. Participants from all levels of physical disability are represented in the requirements capture and video prototyping stages, and usability studies are carried out with able-bodied participants and participants with moderate-to-severe disabilities. Finally, evaluation of the BCI prototypes are carried out with able-bodied people while the sample of people with disabilities is concentrated on those having more severe (major) physical disabilities. See also the general discussion (Section 6.6).

activity rather than as a means of enhancing social interaction between friends and family. (This is contrasted with ratings for a photo browsing application which participants gave in the same questionnaire, for which developing a stronger connection with friends and family was the second reason for desiring the application after the desire to control selection of photos.) Still, one user indicated the desire to be able to compile playlists for friends, which indicates that sharing music might also play a role in a person's social circle.

Interestingly, an important use of a music player would be to control music in the background while doing other tasks. The answers to the questionnaires are reflected in one user's comment that an important feature for him would be that the music player is able to be 'minimized but still controllable (play/pause etc. appear in the task bar; at the bottom right when using windows media player)'. Even for a BCI system, it is possible that a suite of applications would allow a user to select and play music before switching to another application such as sending an email or browsing the Internet.

6.2.3 Implications for Design

Results from the questionnaire indicated that users' goals of having a music player would be to control the music they listened to, and to spend time alone enjoyably. Thus, the development of the prototypes focussed on finding out how users would desire to control

Table 6.2: Scores according to top most desired uses of a music player (Most desired=3 points, 2nd most desired=2 points, 3rd most desired=1 point)

To have control over the music I listen to.	34
To pass the time enjoyably while I am alone.	22
To use music to create different atmospheres or moods.	14
To use music to express my feelings or thoughts.	7
To find new music that I have not seen before.	6
To have a stronger connection with my friends and family through music.	0

a music player given the constraints of BCI control characteristics. In particular, the aim was to to discover what functionality and trade-offs users would be able to accept given the constraints of BCI control characteristics.

6.3 Video prototype and scenarios

While the questionnaire was designed to find out the general goals that users desired the applications to achieve for them, it was necessary to address the specific issues of controlling such an application using a MI-BCI. With such a BCI, error rates and time taken to make a selection can be quite high. In addition, work on being able to control BCIs truly asynchronously is in progress, with current systems unable to stably allow for an 'idle' or non-control state where the user is not trying to give a command to the system. As these are issues that are not common with other input technologies, it is important to obtain feedback from end users about how they might wish to use such a system. Video scenarios were chosen as a method of demonstrating these, partly because it would be the easiest way for AT professionals to demonstrate the prototype without requiring extra training.

6.3.1 Methodology

Video prototypes of the interface were created using cardboard and paper cutouts (here referred to as a paper prototype). As users were likely to be familiar with a scanning system from using other AT systems, a scanning-based system was used to demonstrate some features of the music player.[2] The aim was to show how a music player controlled with an MI-BCI might work, and to highlight the problems with errors and a longer time taken to achieve goals that might be encountered in this system. Thus, although a scanning-based system might not be used in the eventual music player design, it allows the features of the interaction to be easily illustrated and users naive to BCI to understand what was going

[2]Generally, a scanning system is a method of computer interaction where an on screen 'scanner' automatically cycles through selectable objects on the screen, marking them in succession for a set period of time. With a single switch scanner, an interface object is selected when the switch is depressed while being marked for selection. Additional switches add functions such as reversing the direction of the scanning. Examples of scanning interfaces can be found elsewhere, e.g. in Roark et al. (2010) and Biswas and Robinson (2008a).

on. Before any of the music player control options were presented, users were shown a video explaining where the controls and playlist were in the 'music player', and how the scanning system worked with imagining hand movements as in BCI (still frame of video shown in Figure 6.2). 7 participants (1 female; age range 18–46, mean 32.4 years) took part in the feedback sessions in the form of 4 individual interviews and one focus group. The interviews were conducted by BCI researchers in Italy and Germany, while the author conducted the focus group in Italy through an Italian translator who was an AT professional. Note that all participant quotes are interviewers' translations; for the interviews, notes were taken by the interviewer, and subsequently analysed by the author.

6.3.2 Results

Starting and stopping music. To address the issue of there being no true asynchronous control (i.e. a selection to the system would always be made after some period of time), users were shown a video prototype of a music player that started and stopped the music, or skipped to the next or previous track even though the listener was not trying to control anything. When asked if they would use this music player at all, 5 of the 7 participants said that they would accept a system that started and stopped randomly, as long as this did not happen too often as it would quickly become irritating or tiring. However, one of the participants also indicated a desire to be able to stop the BCI if this happened. For the other two participants, this was not an acceptable music player. Participants were also asked whether any of a set of options would be preferable to having the music player start and stop randomly. For each option, a video demonstration was shown to the user and are described as follows:

1. do a sequence of binary selections to activate, or unlock, the player (in this example, left, right, left, left)

2. remove some functions that would abruptly change the music being played more often (for example, skipping to the previous or next track)

3. create the playlist but someone else can decide when to start and stop the music

In response to this question, three of the users would prefer to use a sequence of controls to lock and unlock the music player, while two of them indicated that they would use the BCI to create a playlist and then take the cap off. One person preferred the default, where the music started and stopped randomly, while another would not accept any of the available options. Users also suggested being able to choose when to use the different options.

Playlist generation methods. This part of the sessions aimed to find out how users would choose their music given a range of options and a demonstration of the time it could take to select something to play. Users were shown videos of two commonly used ways of selecting music that are currently in use: iTunes to select an individual track and Moodagent on a

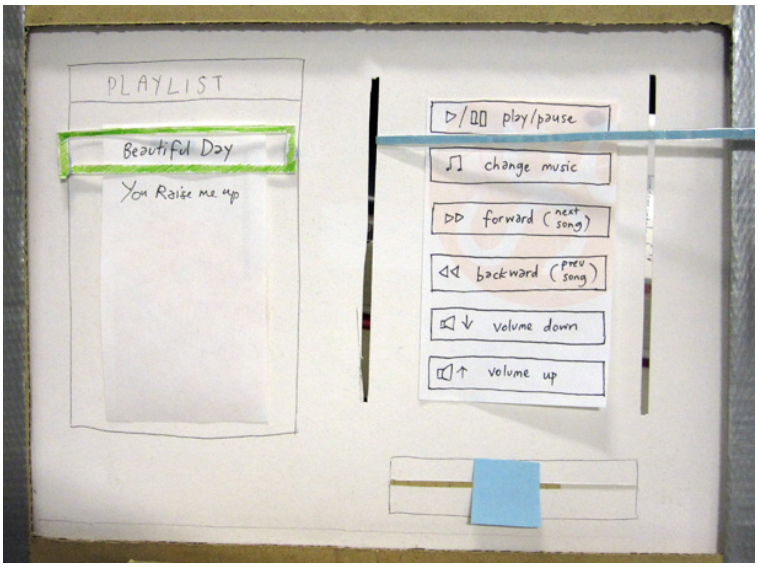


Figure 6.2: Still frame of music browser video prototype for an example video. The 'screen' of the music browser is created with a cardboard frame and paper. The current playlist of songs is the 'panel' on the left. The top panel on the right of the screen is a list of functions such as play/pause, back/next, or change music. A 'scanner' is simulated with a blue strip which is moved vertically as the video progresses. Finally, the bottom-right panel shows the MI-BCI feedback: if the blue feedback bar is pushed to the right, the scanner selects the function at the scanner's position.

mobile phone. All of them would use an iTunes method of choosing their music (searching for individual songs), while 4 of them would use the Moodagent system.

Users were shown several options for generating playlists using a MI-BCI, using the same video prototyping method as in the previous section. Video stills from these options are shown in Figure 6.3. In the first option (left of figure), the system tries to guess the kind of songs the listener may want to listen to by presenting them with one song at a time; the user then makes a binary decision to accept or reject the song. In the second option (middle), the user is presented with a choice of moods or genres, which the person can choose in order for the system to generate a playlist. In the third option, the user is able to select a combination of features by choosing a level for each available feature. The combination is then used to generate an automatic playlist. In the fourth option (right), the user chooses the artist, album or song they desire to listen to with a binary tree menu selection. The expectation was that, on being presented with the slow option of selection with the binary tree menu, participants would prefer options that would return a playlist of songs at a lower cost (i.e. options two and three).

Participants' opinions of the playlist generation options were varied and some did not give an explicit preference for the choices available. For the participant with minor disabilities, all the options seemed good to him but he indicated that he would like to have a choice between different options. One participant who was moderately disabled thought the same

but suggested that for the accept-reject option, sets of songs could be accepted or rejected rather than single songs. Another thought that the accept-reject option might be good for a small collection of songs but difficult for a large collection, liked the mood features, especially the idea of being able to select combinations of features, and thought that it should be possible to make the choice to choose an album. The third participant with moderate disabilities preferred the choice of selecting an album, saying that he preferred to 'choose with consciousness' the music he listens to.

For the participant who was classed with a severe disability, none of the options appealed and he said, 'I would like to add single songs to a playlist by "scanning" all my available songs' in a linear fashion. He also suggested that the system could allow him to easily select songs that he listened to most of the time based on past history. The end user with a major disability found the option to create a playlist based on configuring a number of features 'very very nice'. However, he also indicated that it is 'fundamental' to find a song by searching for the artist, and that it should always be a possibility. Finally, the lady with LiS indicated that the first option was the best. She did not find the mood features acceptable and said that although it was slow, the fourth choice to select individual tracks was preferable as she would want to select songs precisely.

6.3.3 Implications for Design

The desire for choice and customisability amongst users was apparent, as well as the ability to select exactly what music to listen to (at least as an available option). Users showed differing levels of tolerance to imprecision in music selection. For two of the participants, it would be preferable to remove the cap after selecting songs to play. It is somewhat unclear what was meant by this, although one can speculate that the participants would have wanted to select the playlist and start the music playing, before asking someone to remove the cap for them. This would allow control of the choice of music and when it started playing, but not when the music would stop. An alternative view is that, it is much easier to indicate to a human being (such as a friend or carer) the desire to stop music in comparison to what music you would like to listen to. As participants indicated the desire for control and to select exact songs to listen to, in the next stage the aim was to explore and confirm this finding with actual prototypes and explore how users would actually trade off precision of song selection and music player usage with degrading levels of control.

6.4 Prototype study

Given that the priority disabled participants had for a music player was to be able to control the music they listened to, both in terms of functions such as playing music and in creating playlists, these themes were explored further in the context of how preferences, expected or actual use of a minimalistic music player would change as control degraded. In particular, the aims were to find out

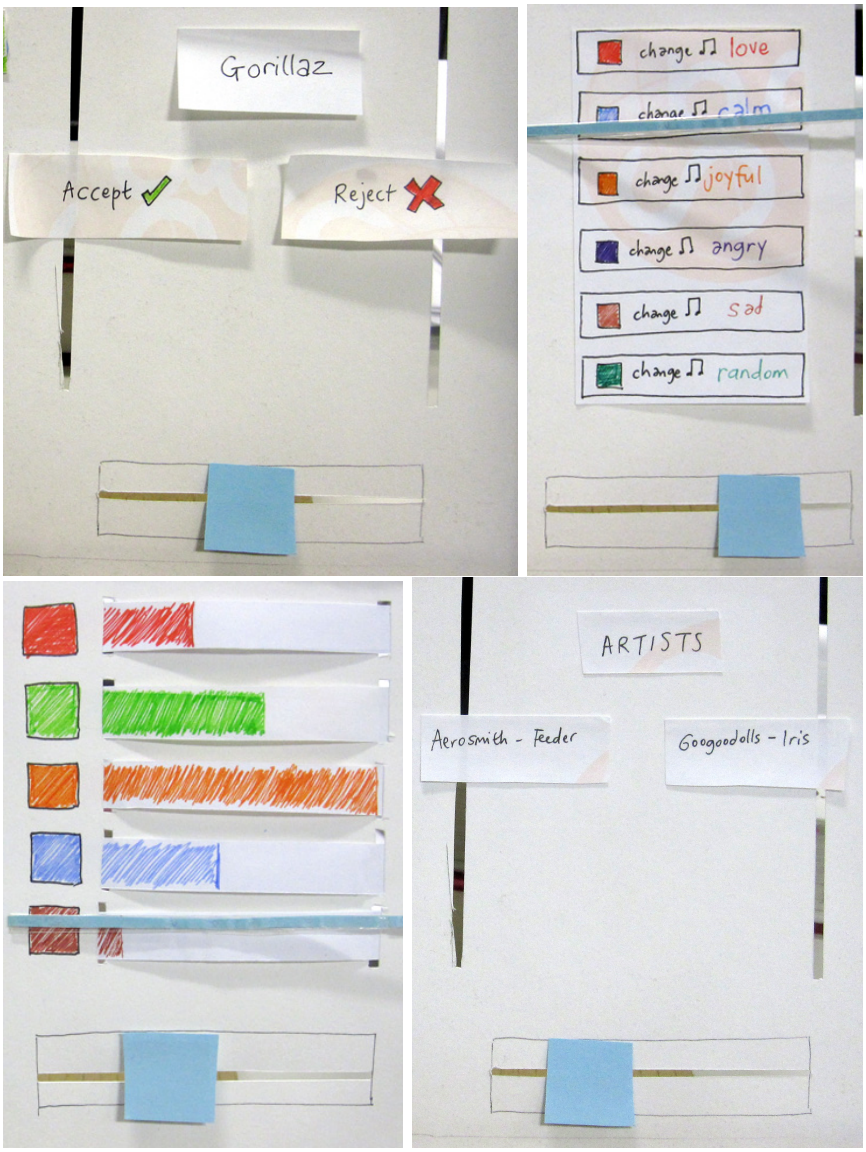


Figure 6.3: Playlist selection options presented to disabled participants. Top-left: The system suggests songs to the user, which are accepted (moving the feedback bar to the left) or rejected (moving the feedback bar to the right) by the user. Top-right: The system presents several moods or genres to the user, which can be selected using a scanning system. Bottom-left: the system allows the user to set a level for each mood or genre, which then provides a combination of choices for the user. Bottom-right: The user selects a specific artist, album or song through a binary tree menu selection.

1. how [expected] use of the system would change as the level of control diminished: when people would simply stop using the system and when they would reduce the time spent listening to music.

2. how people would use the simple mood playlist generation system over selecting specific albums or songs where this would take a slightly longer time to do, and how people would change the way they selected music to play as control degraded. It was expected that with diminishing control, people would prefer to use the random or mood selection methods rather than the album selection, as it would take far less time to generate a playlist of songs to listen to with the former option than with the album selection.

3. what functionality people would notice was missing, or would find unacceptable not to have.

The prototype was tested and evaluated with a variety of users and a variety of input modalities. Firstly, non-disabled participants evaluated the prototype in a single session in a usability lab (Section 6.4.2). Several also went on to evaluate the prototype in a longitudinal study described in Section 6.4.3. Disabled end users evaluated the prototype with a variety of input mechanisms (Section 6.4.4), and finally healthy participants and disabled users evaluated the prototype using real BCI as reported in Section 6.4.5. Finally, the results from the four user studies as well as those from the questionnaires and video prototyping in previous sections are summarised and recommendations for building a music player discussed in Section 6.5. The next section describes the design of the prototype.

6.4.1 Prototype design

As the goal of the music player was to be simple and minimalistic, the number of functions were reduced from the array found in a typical music player to three: play or pause, 'shuffle' which removes the current song from the playlist and rearranges the songs in the current playlist, and 3 different options for choosing music to play. These were Random, Moods and Albums (Figure 6.4). 'Random' returned a playlist of songs randomly selected from the entire music collection. 'Album' took the user into an album browser that was visualised as a CD shop. An album could be selected by successively selecting 'Left' or 'Right' on the wheel according to an alphabetically divided binary tree, until an album was selected. 'Moods' allowed the user to select one mood from four ('Happy', 'Angry', 'Romantic' and 'Calm'). Selecting a mood returned a playlist generated by Moodagent technology as described previously. The motivation for using this technology is similar in BCI and mobile devices. In both settings user input can be slow and frustrating, and typically users do not explore their music collections, but tend to repeatedly listen to a small subset of music. The Moodagent software allows users to explore more of their music collection, without having to enter details about album or track titles or band names. There is a trade-off between low-effort, but easy activation of playlist generators, and the increased effort (significantly

so in the case of BCI) required for precise control of track selection.

The functions of the system were arranged according to the REx selection mechanism which was described in Chapter 5. In this system, the functions were placed in a wheel where spokes of the wheel divided it into segments, one segment per function. The music player was configured to allow for several input modalities (selection modes). In all selection modes, functions could only be selected within the wheel interface. Apart from Mouse mode, the selector was represented by an arrow which could either be rotating clockwise around the centre of the wheel, or extending to select a function corresponding to the segment being pointed at.

- In Mouse mode, segments of the wheel were highlighted when the Mouse pointer hovered over them. To select a function, the Mouse button was clicked within the corresponding segment.

- In Single Switch mode, a single click of a keyboard key or single switch started the arrow rotating 1 second per segment of the wheel and a second click extended the arrow.

- In Two Switch mode, which was intended to be an intermediary to BCI control mode, pressing and holding the left shift key rotated the arrow round, while pressing and holding the right shift key extended the arrow.

- The controls for the BCI Simulator mode were the same as that for Two Switch mode, except that the system was intended to simulate the control of a BCI system where the system is always analysing brain signals and producing input to the application (since there is no idle or 'non-control' state in our current BCI system). Thus, in this mode the arrow would randomly rotate and extend even when there was no keyboard input. The 'lock' function allowed the user to disable all functions apart from the corresponding 'unlock' function, which then had to be selected twice in order to enable the other functions to control the music player. There was also a substantial, random delay as in Quek et al. (2011) and the arrow was set to rotate 2.5 seconds per segment.

6.4.2 Online simulator: lab session with able-bodied participants

Methodology

Participants were 8 able-bodied people aged 23-34 (5 male) who all own and listen to digital music. They were told that the prototype being tested was designed for use with a BCI system, and that the target end user group was people who are severely physically disabled. Initially, users completed a questionnaire about their current listening habits. They were asked to identify their current contexts of listening to music with a request to 'Draw a diagram of your music listening in different contexts'. They were introduced to the system,

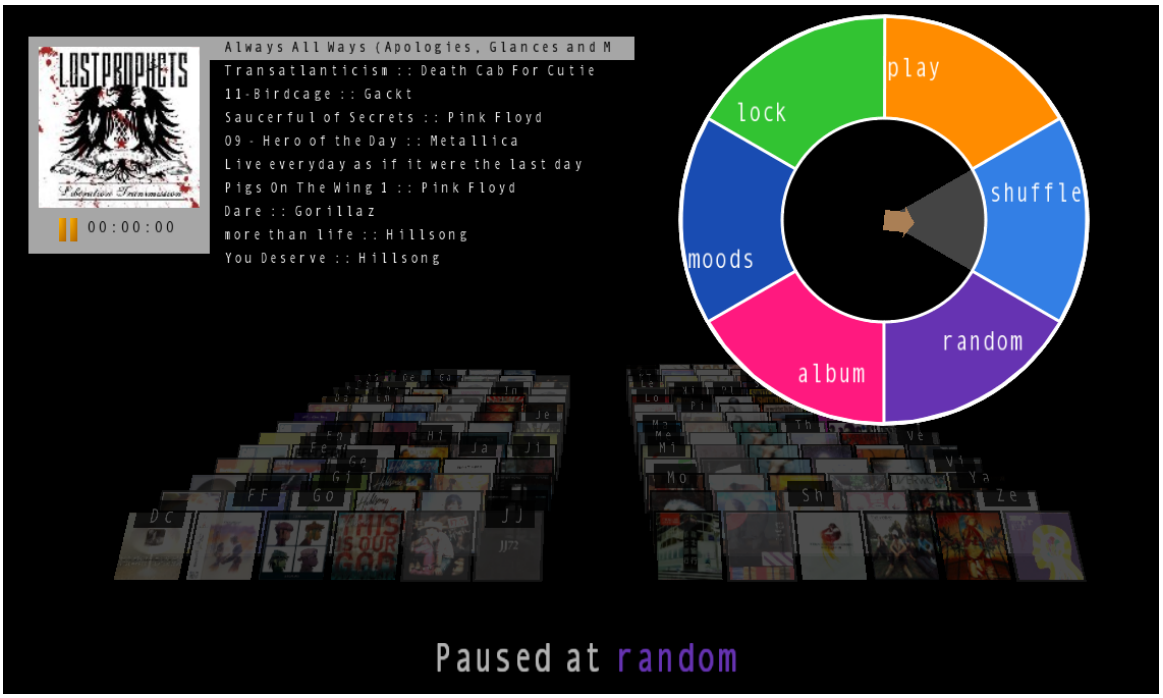


Figure 6.4: The music player design in BCI mode. In a 2-class BCI, one mental state is used to rotate the arrow round the centre of a wheel, while the other mental state is used to extend the arrow to select a segment on the screen.

and for each selection mode (Mouse followed by an intermediate 'Two Switch' mode and then the BCI Simulator), they were asked to carry out some tasks and to provide subjective feedback on the system.

For each user-defined context and input modality, participants were asked to consider a situation where they could use *only* the current music player system (by means of the input modality) to access their music. They were asked to estimate the length of time they would expect to use the music player, and this was compared with their original estimate of the length of time they usually spend listening to their own music. Secondly, participants were asked to estimate the proportion of time they would choose to play music selected using each playlist option. (For example, a person might choose to select music using the Mood selector 10% of the total listening time, 50% of the time with Album and 40% of the time with the Random selector.) They also completed a System Usability Score sheet (SUS) (Bangor et al., 2008), assessing the music player in each input modality. The Microsoft Desirability Toolkit (Benedek and Miner (2002), (henceforth shortened to Desirability Tk)) was chosen as the means for obtaining experiential feedback as it is more engaging than a simple questionnaire, allows users to give qualitative feedback on what they want, and enables the dominant feelings they have about the system to emerge. Users are also encouraged with this method to provide negative feedback on the system (Travis). 80 words were randomised and presented in the form of a table in Microsoft Word. Users were asked to choose 3-5 words to describe their experience with the music player for each input

mode. This formed discussion points with the experimenter.

Results

Contexts. Participants identified between one and four contexts with which they might use a music player on the desktop: four people identified one context, one identified two, two identified three and one identified four. Thus, 16 contexts were identified in general. All participants defined their contexts in relation to life activities such as 'working', 'getting ready for school', or 'studying'. Additionally, two participants explicitly defined some contexts in relation to the desire to create or express moods: 'When I can't focus', 'When I'm sad', 'Creating moods'. A few participants identified contexts for which they would not use a desktop music player such as 'on the bus' and 'gym'; these are not considered in this study.

Overall evaluation. Figure 6.5 (left) shows a summary of the user ratings of the music player based on the SUS for each mode, indicating a decreasing trend from full Mouse control to BCI Simulation control (average scores were 76.25 (±12.4 SD), 65.5 (±6.6 SD), 46.0 (±16.6 SD) for Mouse, Two Switch and BCI Simulator modes respectively).

From the words presented in the Desirability Tk, each user selected 2.9, 1.1 and 0.8 positive words on average for Mouse, Two Switch and BCI Simulator modes respectively and 1.1, 2.0 and 3.4 negative words to describe their experience with the music player. Positive words for the Mouse mode reflected the aesthetic appeal, simplicity and overall ease of use of the interface ('Attractive', 'Easy to use', 'Simple'), and the novelty of the mood functions ('Creative', 'Innovative'). Two participants, s2 and s6, collectively selected the words 'fun', 'entertaining' and 'stimulating' to describe their overall experience with the player. Negative words reflected difficulty in using the album selection, with the chosen words reflecting individual differences in tolerance or perception of difficulty, ranging from no comments at all to 'confusing', 'engaging' (s4, meaning it required concentration), and 'time-consuming' (s1: "Didn't like the album selection - would just directly type in the song if i knew the name... Not complicated - just require me to think too much to select."). 4 participants selected the term 'usable', possibly reflecting neutrality or slight negativity towards the overall perception of usability.

In Two Switch mode, s2 added the word 'appealing' to the list of positive terms, meaning that 'the idea that you can use your brain to control is interesting... but it is time-consuming'. The number of negative words increased, with 5 participants selecting 'time-consuming', and expressing an annoyance that was not previously present with the words 'frustrating' and 'annoying'. Participant s7 chose no positive words, saying that it was 'slow', 'system-oriented', 'stressful' and 'old': it felt like taking a backward leap into the past, where technology is slow and difficult to use. Similarly, s5 selected 'hard to use' and 'too technical'. s6 said it was 'irrelevant', since there would be no need for using such a control in everyday life.

Finally, in BCI Simulator mode, the feeling of unpredictability surfaced with selected words like 'misleading', 'inconsistent', 'insecure', 'uncontrollable' and 'unpredictable'. The number of participants selecting the word 'stressful' increased from 1 to 3. s2 explained that the mental load for using the system was high as one had to 'pay attention to what the system thinks - what inputs it's receiving, like whether it thinks you wanna go left or right, so you need to be constantly paying attention to it, and you need to be thinking to let the key go before your option and what your option is and so on - so high mental workload - a lot more than the previous one (the halfway one).'

Only two participants selected any positive attributes to describe their experience of the music player in BCI Simulator mode. s4 again chose 'appealing' to reflect that the idea of using a BCI was appealing, while s8 chose the words 'motivating', 'desirable' and 'responsive'. The first two adjectives reflected his view of the online simulator as a game: 'the hit and miss thing - coz you didn't get it right, you want to go and do it again. It's kind of like life - like you didn't get an 'A' the first time so you want to try again,' while 'responsive' was chosen to indicate that 'it kinda gets your brain going - have to use your brain to direct it. So instead of playing solitaire to get your head going - you know sometimes when you play mind games to get your head going - you could do that. So that's partly music and partly a game. So that kinda gets you going. Just thinking of how to get it to hit on straight–using-it-wise–because it's always moving.'

Estimated time spent. In almost all contexts, participants thought that the music player in Mouse mode was acceptable enough that they would spend the same length of time playing music with the prototype music player as with their current music playing systems (Figure 6.6(a)). The exception was for s8 with the context 'Creating moods', where he thought that this was the context where he would definitely want to select specific songs to play. In this situation, the mouse mode would not allow creation of a playlist of songs, which in his view would make it difficult or time-consuming to use as normal.

Users estimated their usage (time spent on listening to music using this music player) would decrease from Mouse (median 100%, quartile range (QR) 100.0–100.0%) to Two Switch mode (median 50.0, QR 32.1–100.0%) and a slight decrease from Two Switch to BCI simulator mode in terms of the spread (median 50.0%, QR 25.6–75.0%). There were two contexts for which participants indicated that they would eventually stop using the music player altogether as control degraded. Wilcoxon signed ranks tests between conditions revealed a significant difference between Mouse and Two Switch ($z=0.0$, $p=0.0044$) and Mouse and BCI ($z=1.0$, $p=0.0012$) modes, but not between Two Switch and BCI modes ($z=13.0$, $p=0.3670$).

Estimated use of playlist selection options. Figures 6.7(a) and 6.8(a) show the estimated proportion of time participants thought they would spend using each playlist generation option, as a proportion of the total time they would spend listening to music with the music player. In Mouse mode, 6 out of the 8 participants thought they would use the album selection function at least some of the time, 7 thought they would use the mood

selection at least some of the time, and 7 thought they would use the random selection at least some of the time. 3 out of the 4 participants indicated that they would select their music somewhat differently in different contexts. For example, s1 thought that for relaxing, there was a purpose for choosing music and thus specific songs would be required, whereas during a coffee break anything would do. s6 thought that, when she was sad, she might be more inclined to use the moods functionality, whereas in other contexts she might be more likely to select albums. Participant s4 indicated that, although he could identify different times when he would listen to music, he would choose to select music in the same way. This was similar to s7, who had listed one context as General, as he only listened to music one way.

Overall, participants expected that they would select playlists using progressively less precision as the level of control diminished. This is shown by the increasing trend for Random (median 45%, 55% and 75% for Mouse, Two Switch and BCI Simulator input modalities respectively) and a decrease in the expected use of Mood (22.5%, 10%, 5%) and Album selection (15.0%, 5% and 2.5%; quartile ranges follow trend). Typical reactions to the music player in BCI mode were that 'I would only use random, play, and set to lock. It's too frustrating to do anything else.' (s3). In a few contexts, however, participants actually thought that their use of albums would increase. For example, for the context 'Creating moods', s8 explained that in the Mouse and Two Switch modes he would use Random if the music player could be started up with a specific set of songs. For the BCI mode, however, he would use all three playlist selection options. This was 'because the thing itself is creating a mood. The whole hit and miss thing - that's something that's gonna get you going... the whole delay thing, that's interesting.'

Participants' overall reactions to the mood selection method were mixed. Mostly, a feeling of 'cautious enthusiasm' existed, where there was interest and curiosity but also the need to try it out for themselves. Typical responses included 'Yeah I think I would use it' (s4) and '[it's] something i can try out and see if it does work. But it's not something i would try straightaway. But whether it does work, thats another question' (s8). s8 also expressed skepticism in that the mood player could be difficult to tailor to each individual: 'People's moods and genres are different - what it has for happy might not be what someone else has for happy.' In the end, 7 out of the 8 participants expressed an interest in using the moods functionality. The exception was s7, who explained that he was simply too used to selecting playlists the way he currently does by adding individual tracks.

Functionality. Again, most participants appreciated the simplicity of the music player. For example, s3 commented that 'I like that it's uncluttered. I don't have to scroll down in iTunes. If i'm doing multiple tasks at the same time i don't want a music player where in order to do anything i to have to process more information.' However, she also mentioned that she sometimes wanted to re-listen to a track, and highlighted that this would not be possible with the current music player. s1 also commented that the computer's volume control could be used instead of a volume control in the music player itself.

Responses to degradation of system control. Participants displayed several responses with regard to degradation of system control. For the Two Switch mode, most participants expressed a feeling of the system being slow. Frustration was felt especially when the desired wheel segment was missed, as they would have to wait for the arrow to rotate all the way around again. One participant felt that the system required mental concentration. On the other hand, one participant thought that the idea of controlling the system in this way using thought alone was appealing. At least two participants also explained that after having been exposed to the BCI Simulator mode, the Two Switch mode was perceived as being much more tolerable.

Some interesting responses emerged from observations of participants using the BCI simulator. Several participants exhibited behaviour as if they were playing a game, with phrases like 'Come on come on... come on...' as the wheel was rotating round, and 'yes!' when the correct target was selected. Some enjoyed the 'challenge', while most (6 out of 8) expressed frustration. One participant explained that he would still find the system appealing with this level of control if it was BCI, as it was a form of eyes-free interaction which would be entertaining and useful if he was, for example, driving or walking around. As previously mentioned, one found the system motivating in itself. Still another related the feeling to a person having to use a BCI with the experienced level of control: 'Wow, that's frustrating! I am beginning to feel for people who really have to "control" things in this way!'

Summary of results

The lab study highlighted individual differences in the *preference for* imprecision or uncertainty in terms of the choice of music being played, and the *tolerance of* imprecision or uncertainty in both the choice and control of music. The differences in the tolerance can be seen in participants' responses to the music player as control degraded. Some participants would have given up using the music player at some level, while others did not think the controllability would affect the time spent on using the system. Tolerance and preference for both the controllability of music and the choice of music also appear to depend on context. For situations where focussed concentration was required, it seemed necessary to select and play music with minimum or easy intervention, while in other situations part of the enjoyment might actually be to interact with the music player, or to find songs that had not previously been heard. Thus, the need to tailor a system to individual needs was apparent.

6.4.3 Online simulator: longitudinal use with able-bodied participants

Methodology

Participants in the laboratory session were also asked to install the music player on their own computers to use in their own work or home contexts. Out of these participants, the set up was not possible for one user (s8) due to logistical reasons, s7 did not want to use the player as he was not keen on the mood features and could not accept using the album mode to select songs to play, and two (s4 and s5) started the study but chose to terminate

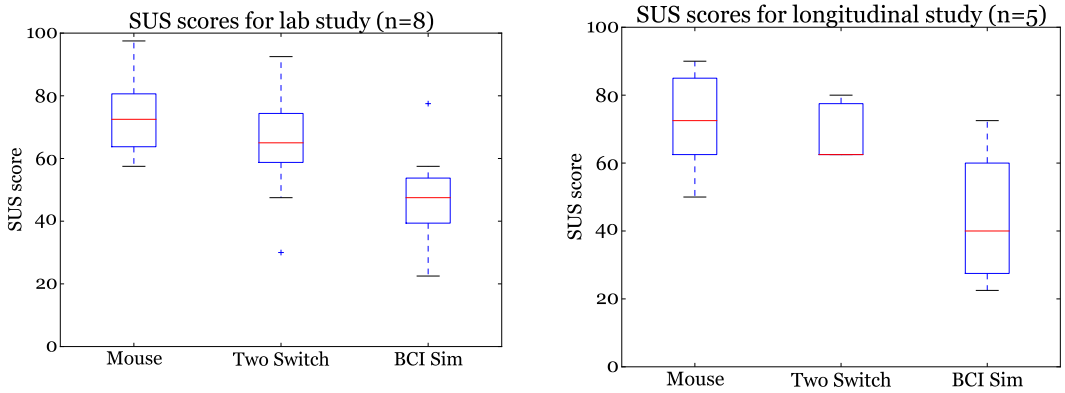
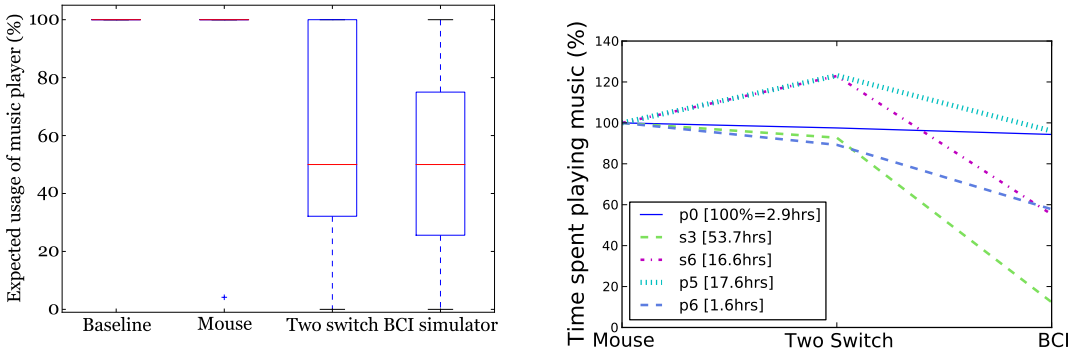


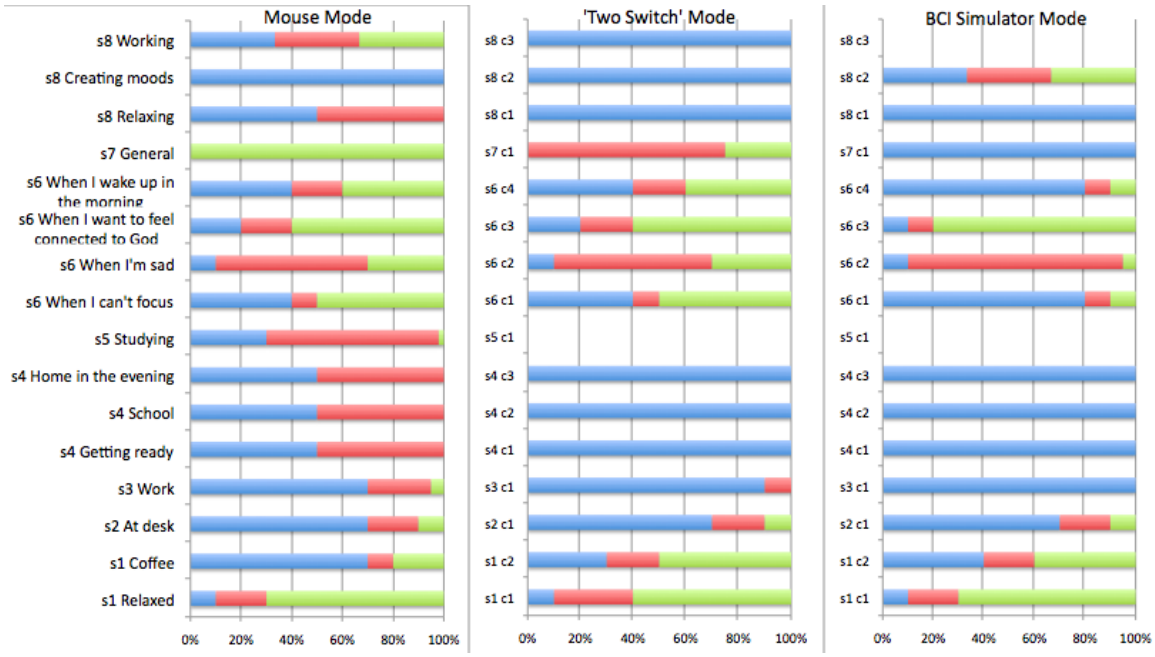
Figure 6.5: User ratings of the system for the lab study (left) and the longitudinal study (right) in different input modalities (Mouse, Two Switch and BCI Simulator) as measured by the System Usability Scale (SUS). A rating below 60 is generally considered to be a bad score, while a rating above 80 is generally considered to be a good score (Bangor et al., 2008). Ratings are comparable for the two conditions, with a decreasing trend in ratings as the control degrades. Note that in the longitudinal study, the scores for participant s1 were excluded as he did not use the music player in the Two Switch mode.



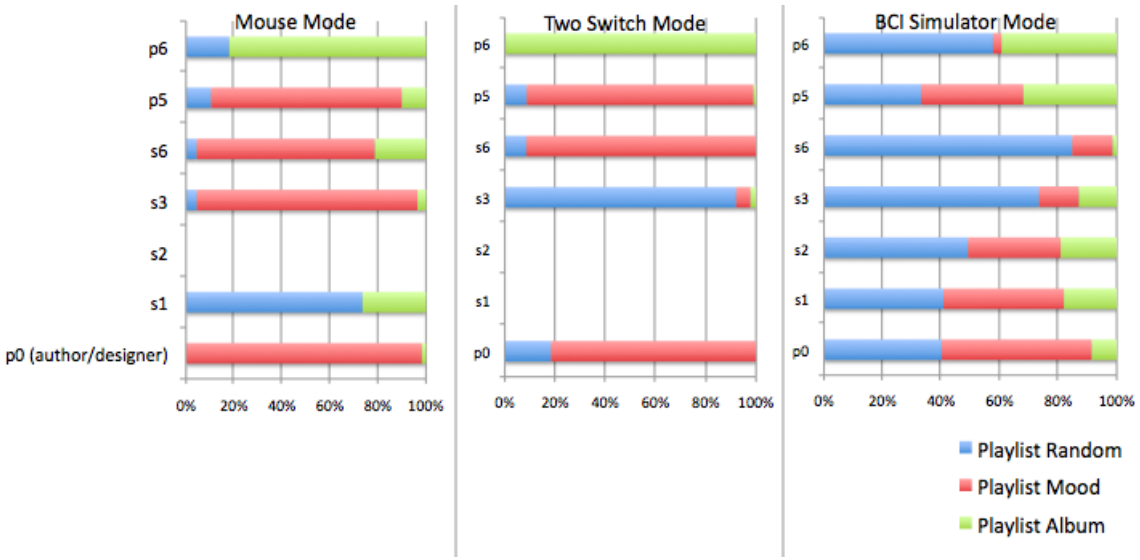
(a) lab (n=8)

(b) longitudinal (n=5 of 7)

Figure 6.6: Percentage of time participants would spend ((a)) and actually spent ((b)) listening to music with the music player for the input modalities Mouse, Two Switch and BCI Simulator. For the lab study, participants' normal usage is taken as the self-reported baseline of 100%, while for the longitudinal study, the baseline of 100% is the time spent playing music in the Mouse mode. Note that in the longitudinal study, results from 4 out of 6 participants are shown as data for the other participants was incomplete. p0 is the author (designer). Table 6.3 provides additional information on the order and length of use of the different input modalities.

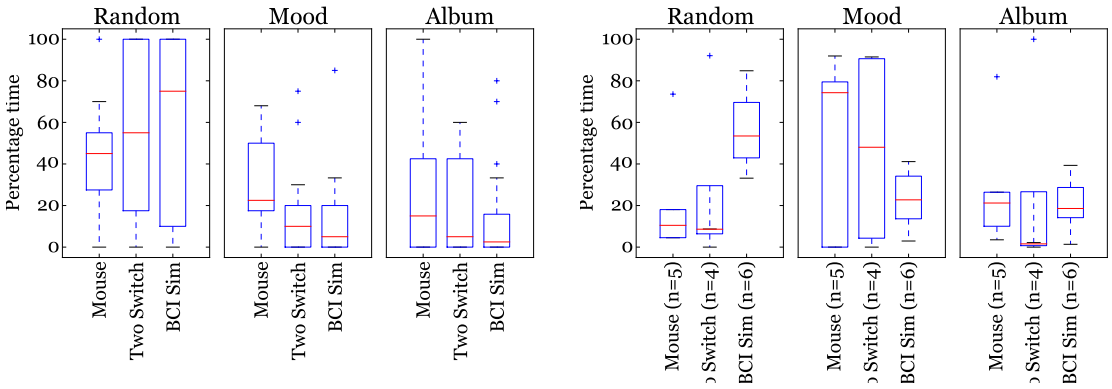


(a) lab study showing expected usage in different contexts (n=8). Blank rows indicate that the participant would not use the music player in this context and input modality.



(b) longitudinal study (n=6 + author/designer) showing actual usage. Blank rows indicate incomplete data collection (see text for details).

Figure 6.7: Participants' expected ((a)) and actual ((b)) usage of playlist selection options for the lab and longitudinal studies respectively. The use of each option (random, mood and album) is expressed as a proportion of the total time one chooses to play music. The studies compare how participants would choose to select music using the input modalities Mouse, Two Switch and BCI Simulator.



(a) lab (Expected usage)

(b) longitudinal (Actual usage)

Figure 6.8: Box plots of participants' usage of playlist selection options for three input modalities. In the lab study (n=8), participants estimated how much time they would likely use the music player. Usage of playlist options for Mouse (n=5), 'Two Switch' (n=4) and BCI Simulator (n=6) for the longitudinal study show the actual usage (a similar spread of results is found if only the 4 Two Switch data points are plotted). Each Box plot shows the distribution of the time spent in a playlist selection option as a percentage of the total time spent playing music (i.e. 'time spent in a playlist selection option' is considered when the music being played was generated using the particular playlist selection option).

participation after using the music player for two days. Finally, two additional participants (p5 and p6, both male) took part in the longitudinal study. Thus, usage statistics for the user study are reported for 6 participants plus p0, the author/designer who had much previous use of the system.

Similar to the lab session, participants were asked to use the music player in the three different input modalities, with the aim of finding out how the choice of playlist would differ between the input modalities. Because the aim was to examine how the use of the system changes as the control is degraded, each participant started off using the music player with the Mouse input modality. The length of the study and way the input modalities changed were configured for each individual differently. Table 6.3 shows a summary of the length of time of the study and the way in which the input modalities were cycled through. Where the participant cycled through the input modalities more than once, the system automatically switched input modes when the allocated period of time for a modality had passed. Otherwise, an email was sent out to the participant inviting them to manually switch to the next mode at the end of the day. The table also shows the number of tracks that participants used for the duration of the time. As can be seen, some participants chose to use their entire music collection while others chose to use only a subset of their collection for the study.

Usage statistics were collected by logging the time spent playing music in each playlist mode.

Although an attempt to log participants' context was also made, in the end, the study was too short to make any conclusions about the relationship between context, input modality and usage of playlist options. Thus, the results are reported without regard to the user's context. Participants to the lab study were asked to complete the same questionnaire as the lab session before switching to the next input mode. Finally, a post-study interview was carried out with participants after the completion of the study.

Table 6.3: Length of time of study and order of conditions (input modalities Mouse, Two Switch (TS) and BCI Simulator (BCI Sim)) for participants completing a longitudinal study using the music player in their home or work contexts.

Participant (Gender)	Total days' usage	Order of conditions (input modes)	Num Albums / Tracks
p0 (author/ designer) (F)	18	cycle of 3 days mouse, 3 days BCI Sim, 3 days TS	398 / 4003
s1 (M)	4	2 days mouse, 2 days BCI Sim	5 / 6
s2 (M)	6	2 days mouse, 2 days TS, 2 days BCI Sim	714 / 7867
s3 (F)	6	3 days mouse, 3 days TS, 3 days BCI Sim	76 / 805
s6 (F)	9	3 days mouse, 3 days BCI Sim, 3 days TS	280 / 4409
p5 (M)	12	cycle of 2 days mouse, 2 days TS, 2 days BCI Sim	47 / 541
p6 (M)	9	3 days mouse, 3 days TS, 3 days BCI Sim	46 / 489

Results

Overall evaluation. Figure 6.5 (right) shows the spread of participants' overall SUS ratings of the music player in the different modalities. Similar to the lab study, there was a decrease in the average score from Mouse (72.0 ±14.6 SD) to Two Switch (69.0 ±8.0 SD) to BCI Simulator (44.5 ±19.0 SD) modes. On average, participants chose 3.2, 1.2 and 0.3 positive words and 0.2, 0.6 and 3.0 negative words to describe the music player in Mouse mode, Two Switch mode and BCI modes respectively.

For the Mouse mode, hedonic positive words fell into categories describing the aesthetics ('clean', 'bright', 'new'), being 'creative' and 'satisfying', 'appealing' and 'approachable'. The most common positive words referring to usability were 'consistent', 'responsive' and 'easy to use'. Other words included 'predictable' and 'time-saving'. The words 'frustrating' and 'time-consuming' were chosen to reflect users' difficulty in selecting albums using the binary selection. For Two Switch mode, added positive words included 'creative', 'stimulating' and 'fun' (s6). The positive words used to describe usability were 'reliable' and 'accessible'. The main negative words used to describe the music player in the Two Switch mode were 'slow' (3 out of 5 participants), 'annoying' and 'frustrating'. Finally, for the BCI Simulator mode positive words were 'creative' and 'impressive'. The most common negative words were 'slow' and 'frustrating' (4 out of 6 participants), followed by 'time-consuming'

and 'inconsistent'. Other words were 'hard to use', 'uncontrollable', 'complex', 'misleading' and 'boring'.

Time spent listening to music. Figure 6.6(b) shows the time spent on using the music player for each input modality over the allocated days, with reference to the Mouse mode (taken to be 100%). Results are shown for four participants plus the author/designer. The general trend was that the level of use of the music player between the Mouse and Two Switch modes (mean 107.0% ±16.1 SD over four participants) was comparable, while that of the BCI simulator mode decreased (mean 55% ±29.6 SD). As well as the author, the participant who had a longer use of the music player and switched between the different input modalities (p5), did not show a substantial decrease in the use of the music player in BCI mode.

Usage of playlist selection options. Figures 6.7(b) and 6.8(b) show the time spent playing music in each playlist option as a proportion of the total time spent playing music, for each input modality. For both Mouse and Two Switch mode, the Mood selector had the highest usage in terms of the median estimates (median 74.3%, quartile range 0.0–79.5% for Mouse mode; med 48.0%, QR 4.3–90.6% for Two Switch mode). For the BCI Simulator mode, the use of the random selector increased (med 53.5%, QR 42.9–69.6%) while that of the mood selector decreased (med 22.7%, QR 13.6–34.1%). Interestingly, the use of album selector was comparable for both Mouse and BCI Simulator mode (med 21.2%, QR 10.0–26.4% for Mouse; med 18.6%, QR 14.2–28.7% for BCI Sim), but was reduced for Two Switch mode (med 1.6%, QR 0.7–26.6%).

Due to the small sample size and length of time of the study, conclusions cannot be drawn and the reasoning behind participants' choices cannot be made completely clear. Some of the smaller percentages could be due, for example, to participants simply trying out the playlist option before deciding that it was suitable for use with the input modality. However, it is interesting that some participants continued to use, or used the album selection option more than in other modes. p5 indicated that in some contexts (e.g. where focussed attention was required), it was easier to invest time selecting an album initially as it would allow one to play music that was known to be what was needed at the time. This was also the experience of the author. However, s3 and p6 said that that, while they may have done this if the music was played reliably, the unpredictability of when the music player would unlock itself meant that it was not worthwhile to invest the initial time at the beginning.

Interestingly, the album selector was used the least in the Two Switch mode rather than in the BCI Simulator mode. It is possible that this phenomenon is simply due to sampling error, as data from only 4 participants is included in this input modality. Certainly, p6 continued to use the album selector 100% of the time in Two Switch mode and only 40% in BCI Simulator mode. On the other hand, user comments indicated that there might be a logical reason for not choosing to use the album selector in Two Switch mode. In the random and mood selection modes, participants liked being able to click 'shuffle' easily, which was slower to do in the Two Switch mode and even slower in the BCI simulator mode.

It is possible that with the Two Switch mode, selecting 'shuffle' often is still perceived as a less costly transaction than selecting an album. Thus, an interaction between the playlist selection options and input modalities may be seen to exist. Album selection in Mouse mode could be somewhat tolerated because it was quick enough to do; in Two Switch mode it was too slow and not worth the hassle (for some participants), while in extreme loss of control as in the BCI Simulator mode it was a way of ensuring that a good selection of songs was chosen. The finding that p6 chose to use the random selector in the Mouse mode but not in the Two Switch mode would fit into this account: since his tolerance of randomness in song selection is low relative to other participants, it follows that in the Two Switch mode, he might seek to eliminate all uncertainty so as to limit interaction with the music player as much as possible while listening to music.

The order of exposure to the different input modalities appear to have an effect on the playlist selection options. For example, the moods functions continued to be highly used in the Two Switch mode where users were either exposed to different modes on more than one occasion (p5) or the Two Switch mode after BCI Simulator mode (s6). On the other hand, s3's usage of the moods playlist selector decreased substantially on exposure to the Two Switch mode. During the post-study interviews, she indicated that she liked the moods function in the Mouse and Two Switch modes, and suggested that if she were to use the Two Switch mode again, taking a slightly longer time to select a mood and to shuffle tracks would not seem as tedious as it did before. s2 also revealed that while he was using the Two Switch mode, it had seemed 'incredibly sluggish, but nothing compared to BCI Simulator mode'.

Participants revealed different experiences of the playlist options. As in the lab study, all users found the album selection difficult to use. Nevertheless, participant s2 continued to express his liking for the binary album selection even after the study: 'I really liked the album selection, it was funky. It forced me to think about the albums, what the names were and stuff. I'm not sure if it's just 'cos i'm a computing scientist though... just the binary search was pretty cool. 'Cos it was $log(n)$ complexity so that was good.' p6 indicated that he was surprised by how different it was to have the tracks within an album being played in a random order, as he was usually religious about playing them in the correct order. Most participants mentioned that they used the random function more as the interaction with the music player became more time consuming. Finally, the most interesting and diverse responses regarded the mood selector. On the one extreme, participant p6 did not like it at all, saying that he did not find it different from random and preferred to select exactly what he wanted to listen to anyway. On the other extreme, s6 liked the mood selector, saying that it 'suits my taste very well'. s2 found that the 'calm' mood filtered out songs that that she did not like, or that were not appropriate for working, but was not entirely convinced about the choice for some of the other moods ('I did try some of the others but i wasn't quite sure what angry was supposed to mean so...'). In total, 4 out of the 6 participants expressed a liking for and used the moods function, although some mentioned that in slower control modes it took too long to select.

Functionality. Most participants who completed the study expressed that the music player contained the functions they desired in a music player. Two participants again appreciated the simplicity and lack of clutter of the music player as compared to iTunes. However, two mentioned that volume control would have been good to have, one wanted the functionality to skip tracks without shuffling (a default setting), one wanted the music player to start at the last place it stopped, and one indicated that the lack of a repeat was frustrating.

Responses to degradation of system control. Participants exhibited several responses with regard to degradation of system control. Similarly to the lab study, participants found the Two Switch mode to be rather slow, but upon being exposed to the BCI Simulator mode, expressed that they should probably have been more tolerant of it on hindsight. s6 expressed that it was 'annoying at the start because I had to turn the arrow... but at the end I found it quite manageable and unique.' It is possible to speculate that after a while, the rhythm of the rotation and selection required in the Two Switch mode becomes relaxing and no longer a source of frustration. Relatedly, for both Two Switch and BCI Sim mode, p5 expressed that it felt like he was being 'unnecessarily frustrated': 'I just thought, "But i was able to control it easily 2 days ago!"'

In all the comments for the BCI Simulator mode, frustration was also expressed. For example, s3 wrote, 'It was extremely frustrating and stressful to be interrupted from work because the player had changed the settings (paused the music or changed to a different setting) and to have to then go through the selection process again. On a number of occasions, I lost patience with re-setting the settings, and resorted to workarounds like relaunching the player so that I could simply select random play and lock (which meant fewer attempts [selects] to get the right option selected). I also used the mute on the computer once or twice when there were songs that were too distracting to work to. Basically, even if I was sufficiently incapacitated to use a system like this, I think that most of the time, Id prefer to have someone else choose my music, or just have it playing all the time.' On the other hand, when asked if she would use the BCI mode if she was disabled, s6's response was 'No because it would be exhausting if i were disabled. But i think if i were disabled and i really liked music, i would still work to use it.'

Summary of results

The findings from the longitudinal study largely follow and enhance those of the lab study showing that individual differences in tolerance and preference for controllability and choice of music exist. An additional finding is that the biggest source of frustration in the BCI Simulator mode was that the music player would unlock itself and perform an unintended action every so often. An interesting observation was that a few users found value in investing time in selecting the right kind of music as interaction with the music player became very costly; however, the unpredictability of the system while playing music lessened the value of doing this and participants either gave up using the system, reduced use, sought to 'fight with' the player or tolerated it (let the music player play anything it wanted). Thus, the main limitation of the control was the unpredictability caused by the lack of 'idle' state.

The other main issue participants found was that selecting anything (i.e. pausing or skipping track) with the BCI Simulator took far longer than in the Two Switch mode as the system had to be unlocked before selecting any of the functions. The conclusion that these additional qualities in the degradation of control had a larger effect on participants' use of the system in BCI Simulator mode than the input simply being slower is backed up by the fact that the time spent using the Two Switch mode was comparable to that of the Mouse mode, while in most cases the usage of the BCI Simulator mode decreased.

Another problem highlighted by participants was that it was too time-consuming to select albums. This shows that the mood selector in itself was not adequate for controlling a music player, highlighting a need for improvement of the album selection mechanism. Still, there were participants who used and liked the mood function in Mouse mode, however degrading the control made them switch to using random at some point instead as the design of the system meant that random was easier to select than a mood. This suggests that if the mood function was as easily accessible as random, it may have allowed users to continue selecting mood. Finally, the features volume, repeat and history were highlighted as necessary functions that were missing from the player.

It is worth noting that results should be taken with caution due to the small sample size and relatively uncontrolled ordering of the experimental conditions and duration of the experiment for each participant. Nevertheless, the present findings are valuable for the purpose of making recommendations for design. Firmer conclusions about the specific sources of frustration, and individual differences in the level of tolerance and preferences regarding uncertainty in music selection, may be drawn with a more tightly controlled study involving a larger number of participants.

6.4.4 Various input modalities and settings: case studies of disabled participants' use

4 people with varying degrees of disability were asked to evaluate the music player with various input modalities in different settings. In this section, experiences and user comments with the prototype study are reported, with a summary and implications for design at the end of the section. Note that all names have been changed for confidentiality.

Paulo (Moderate disability)

Paulo is a 22 year old male from Italy who was involved in an accident a few years ago. As a result, he is unable to grasp objects, but uses a Mouse and keyboard with both hands and gets around in a manual controlled wheelchair. He is very bright and motivated as shown by his latest project to program a robot. Paulo was introduced to the music player using both mouse and single switch functionality, but preferred to use a mouse. The Desirability Tk was used to engage Paulo in expressing his initial impressions of the music player.

Current music listening habits and contexts. Paulo indicated that he listens to music in two contexts: when he is focusing on work, and at all other times. He indicated differences in how he usually listens to music: When focusing on work, he selects an album to listen to most of the time, while all other times, he mostly uses smart playlists.

Initial impressions of the music player. The adjectives that Paulo selected to describe the music player were 'Predictable', meaning that the control input of the single switch was controllable, 'Accessible', and 'Simple' meaning that in general the system was understandable, easy to use and intuitive. However, he said that he would probably not use the music player in every day life as it is too difficult to select an album to listen to (selected Somewhat Disagree, on a scale of 1-5, 1 being Strongly Disagree and 5 being Strongly Agree).

Fred (Moderate disability)

Fred is a 52 year old composer from Switzerland who has Guillain-Barr syndrome and was undergoing physiotherapy at the time of study. Fred is able to use his hands, but not to grasp objects, and uses a manually controlled wheelchair to move around. The author invited him to evaluate the music player after a weekly physiotherapy session at the rehabilitation hospital.

Current music listening habits and contexts. Fred indicated that he only listens to classical music or French songs, and usually enjoys music during dinner time either through CDs or standard music players on a desktop computer. Since sound quality is of utmost importance to him, his digital music collection contains only .wav files. He is accustomed to choosing specific songs to play, and sometimes creates playlists for friends. In addressing the kind of music he listens to, he indicated that music that uses a single instrument is the best as it is more personal and intricate, while if there is company, Mozart or some other symphony is selected as the music is not as technical.

Initial impressions of the music player. Upon being introduced to the music player and using it with a mouse, Fred indicated that it was cumbersome to use the album selector, and suggested that it could be interesting if all the music could be put on a list and rotate around the wheel; the user could click when the desired artist or album passed the selector. This would be equivalent of a scanning system with an original visualisation. Fred was also interested in the moods selector, but expressed a desire to have more precision or customisation than the music player would allow. He wanted to have a hierarchy of moods, where you could choose a mood (for example, 'calm'), followed by an artist or style (e.g. 'Mozart', or 'classical'). On explaining that this could be done with the more complex feature selection system, he commented 'This is nice.' Lastly, Fred indicated that he would like to have the functionality of building up his own library of moods manually by going through the list of music and dropping it into self-defined categories such as 'happy' or 'angry'.

Brenda (Severe disability)

Brenda is an Italian lady who has cerebral palsy and uses an array of switches as input into a computer. She is able to communicate in English verbally. She obtained a University degree in music, and is very experienced with a variety of ATs. The author visited Brenda at her home with an AT professional. Following a brief interview about her current music listening habits, and an introduction to the music player, the software was successfully installed onto her desktop computer. Brenda used a single switch to control the music player, as it was the easiest means for her to provide input into a computer. She was able to complete all the tasks using this input mode.

Current music listening habits and contexts. Brenda identified her contexts of listening to music as 'Working', 'Having friends over', 'Relaxing' and 'When I want to sing very loudly'. In all contexts, she indicated that she listens to music on Random 90% of the time, and for the other 10% she chooses music by her favourite artists or singers. She often creates playlists of music by choosing songs to play.

Initial impressions of the music player. Brenda indicated a strong interest in the music player, saying that she appreciated the simplicity of the mood features and the interface, and she thought that the REx paradigm was very innovative by current AT standards. The words she used to describe the music player, as selected from the Desirability Tk, were that it is 'Time-saving' as it takes her less time to choose music to play, 'Innovative', 'Attractive', 'Energetic' or 'Exciting', and 'High Quality'. She indicated that she would use the mood-based playlist selection method a lot for her own pleasure after the application had been installed on her own computer, and would also try to become more comfortable with using the album selection method. Brenda was happy that she had the ability to fine-tune the parameters of the mood-based playlist selection algorithm, and described the ability to simply pick a mood to obtain a particular type of music as 'time saving'.

James (Moderate disability)

One participant downloaded and installed the music player in his home computer and was asked to evaluate the music player in mouse and single switch mode. James is almost tetraplegic but able to move his hands and use a mouse and keyboard. As he is lives in Northern Ireland, we communicated through email. The procedure was similar to that carried out with non-disabled participants in the longitudinal study in the previous section. Prior to using the music player, James was asked to fill in the initial questionnaire about his music listening habits, and to identify his contexts of listening to music. After using the music player for a few days, he was asked to estimate his use of the music player, and to fill in a post-study questionnaire.

Current music listening habits and contexts. James indicated that he listens to music in three contexts: '1 is late at night in bed, I have a music system in my bedroom and leave the timer switch set so that I can drift off to sleep while listening to a CD or sometimes

the radio; 2 is when I am doing my devotions (daily spell of bible study and prayer), I have praise/worship music on while I do that; and 3 is the most common, when I have music on to listen to while I'm doing something else (usually surfing the web or reading).'

Evaluation of the music player. Unfortunately, the system could not be set up such that James could listen to his own music on the music player, or to obtain any usage data due to technical reasons. Thus, he downloaded and used the sample set of music that was used for the lab and demo sessions, and only the subjective evaluations are reported here.

In terms of contexts, James found that he could not listen to music using the current music player for the contexts 'drifting off to sleep', because the music player lacked a timer to turn itself off after he fell asleep, and 'doing my devotions' because 'the selection of music didn't really include anything suitable'. Thus, he only used the music player in the context 'while doing something else'. For the Desirability Tk, the words 'appealing', 'bright', 'fresh' and 'easy to use' were selected, with the comment that they were 'chosen to reflect highly positive experience of using music player'.

With regard to the use of different playlist selection options, James estimated using the random selector 20% of the time, mood 35% of the time, and album 45% of the time. He also said that he liked the binary selection of albums, and that in the end this was his preferred option as 'takes less time/effort get the music I want playing this way', 'i like being able to control exactly what was playing', and 'its what im most used to' (pre-defined options). He also mentioned that the music player lacked a 'repeat' function that would allow him to replay a whole album or song. He preferred the Mouse mode to single switch, as the switch mode 'took too long and was frustrating' while the mouse mode was 'quick and convenient'.

Summary of results

In terms of overall impressions, the 3 participants who chose words from the Desirability Tk selected no negative words and an average of 4.6 positive words to describe their experience with the music player. Some participants were more enthusiastic about the moods functionality than others; at least 3 of the 4 participants expressed an interest in using moods in some way. One participant was also excited at the prospect of engaging with the mood technology and fine-tuning the parameters to suit her needs. It was clear that participants also wanted the ability to choose individual albums and tracks, such that one participant indicated that he would not use the music player in his own time because it was difficult to select an album, relative to his current solutions. The usability of the album selection was a problem for 2 of the participants; however one said that he liked the binary selection. The repeat function was identified as a missing function of the music player.

6.4.5 BCI study

Methodology

6 non-disabled and 4 disabled (all male) participants tested and evaluated the music player with real BCI. After setting up the BCI cap, the standard calibration tasks were carried out in order to gauge participants' performance. To evaluate the music player, they were given several tasks to complete, followed by a post-study questionnaire to explore how users perceived the interface. One disabled participant also evaluated the player using the Desirability Tk. The results focus on participants' preference for the different playlist selection methods; results from the three disabled participants are reported as case studies.

Results

Able-bodied participants. Out of the able-bodied participants, most indicated that they would mainly choose to use the mood (3 out of 6 participants) or random (2 participants) function to select music if they were using the BCI music player. From a list of pre-defined options, the reasons selected for participants' choice included that this allowed them to take less time and effort to select music, and that they would enjoy listening to music they wouldn't normally listen to. The remaining participant indicated that he would choose to use the album selection as he wanted to be able to explore and find out exactly what was in the music player collection, and he liked being able to control exactly what was being played.

Participant L00 (severe disability). L00 is a music enthusiast who has paraplegia and a limited range of arm movement and grasping ability. He reported having around 3500 songs and 120 albums in his digital music collection, and usually creates playlists by genre, random selection, or shuffling albums. Prior to the BCI experiment, L00 was introduced to the music player using a single switch device. In answer to the BCI mode, he preferred to 'select a mood and then let the system select the songs', as he would enjoy listening to music different from what he was used to.

Participant L01 (major disability). L01 is tetraplegic and does not currently have access to music on his own. Occasionally, he requests music to play but usually his wife chooses the music. He indicated that before the accident, he would choose CDs to play in the car, but currently he has no way of doing so. In any event, in his current lifestyle, L01 feels that he does not have time to think about the issue of being able to play music and until the user study had not really thought about having a music player; thus a music player would be a 'nice to have' feature of his life rather than an essential part.

During the music browser testing, end user L01 became rather excited that he might be able to select a particular artist, Rammstein. However, as this particular artist was not in the sample music collection, he settled for selecting an album by the band AC/DC. The sequence L01 decided on his own was to select the album, play, lock the music player, listen

to the track for some time, unlock the player, select 'shuffle' to change tracks and lock the music player again. On completing this sequence of actions with 100% accuracy (13/13 selections in total), he appeared to be delighted and commented that being able to do this was 'cheering [him] up more than the other BCI trials'.

From the Desirability Tk, L01 selected the terms 'Easy to use/ Simple' ('When I use it for some time, it's easy for me'), 'Desirable' ('From so many albums I can quickly choose [the one I want]'), and 'Advanced' ('Cutting edge, state of the art technology future oriented'). In the post study questionnaire, L01 indicated that he would prefer to use the album selection as he preferred to select the exact album or song he wanted, and that it was what he was most used to. Finally, in terms of functionality L01 pointed out that the music player lacked a volume control function.

Participant L02 (Paulo from previous section). L02 had no strong preference for any individual playlist selection method, indicating that he would choose to use a combination of all 3 selection methods.

Participant L03 (severe disability). L03 is an 18 year old who has muscular dystrophy (Duchenne) and is unable to perform the most common activities of daily living without the help of one of his caregivers. He is able to speak and has residual movement of the hands and fingers. He is able to move around on a joystick-controlled power wheelchair. Although he uses a desktop PC via a trackball for communication with his sisters via Facebook and Skype, and uses the iPhone to manage domotic solutions within his home environment, he does not regularly use a music player on these devices. As such, although he was able to use the music player with a high level of accuracy, he did not have a strong opinion preference for any of the music player options, abstaining from answering the question on which of the three playlist selection methods he would prefer to use. However, he did rank the options in a later question (album=1, moods=2, random=3).

Summary of results

Of the two participants who were moderately disabled, one had no particular preference for a particular playlist selection method, and the other would use the moods functionality. The participant with severe disability preferred the album selector when pressed for a vote, and the participant with major disability preferred the album selector as he was used to choosing his own music, and liked being able to choose exactly what was playing. In contrast, only one able-bodied participant indicated that he would prefer to use the album selection, while the others preferred using the random and moods functions. Volume control was highlighted by one participant as being a missing feature of control.

6.5 Summary of findings and implications for design

In total, feedback from 15 non-disabled and 22 disabled participants was captured in six settings as described in previous sections. The goal was to investigate how people might choose to interact with a music player given the limited constraints of a BCI, in order to provide recommendations for how to design a MI-BCI-controlled music player for someone with LiS. In this section, the results are combined and presented in terms of the aims set out at the beginning of the prototype study (Section 6.4).

Overall use of music player as control degrades. The controllability of a software system depends on the user's physical ability, the choice of input modality and the design of the software application. Physical ability limits the choice of input modalities available, while each input modality has its particular constraints. In the case MI-BCI control, the user is restricted to two inputs, the time taken to select an interface object is very slow, and there is likely to be a degree of error in the interface. The prototype developed in Section 6.4 allowed for a variety of input modalities with which to engage both able-bodied and disabled participants, and an attempt was made to simulate disability by degrading user control using two-switch input and an online BCI simulator. It is worth noting that the music player forced users to use a binary selection to select albums, and as such this can be seen as an initial restriction of control, as it would generally take a user more time to select an album using a standard music player application if the mouse and keyboard are used.

The online simulation studies revealed two continuums which can be used to describe the behaviours resulting in degradation of control. The first defines the control strategy one employs to use the system. Users might choose to simply tolerate the behaviour of the system or change strategies or the way in which they use it (e.g. changing the way they select music). The second relates to the time spent using the system: users might continue to use the system as much as they did before, reduce the time spent using the system, or stop using it altogether. It could be seen that individual differences exist in terms of the degree of tolerance people have, which influences their behaviour, continued use or acceptance of technology as the level of control degrades. Firstly, there was individual variation in when people stopped or would stop using the music player. Some participants would not use the music player at all, for some the threshold was the Mouse or Two Switch mode, others continued to use the player in BCI Simulator mode for a reduced period of time and still others would continue or continued to use the player for the same length of time as the baseline. This reflects the variation in disabled participants' responses to the video prototypes, as well as their responses to the music player prototype as controlled using different input modalities.

Feedback from non-disabled participants in the longitudinal study revealed that, in BCI Simulator mode, the biggest source of frustration was the uncertainty of when the music player would unlock itself and do something random. This was pre-empted as disabled

participants were asked to consider how this might affect the way they controlled the music player during the video prototype study. It was seen that some participants would not use a music player that started and stopped randomly, others would only create playlists then take the BCI cap off, and still others would tolerate the randomness as long as it did not occur too often. Because of the rather strong responses from able-bodied participants in response to this, the recommendation is to eliminate the lack of idle state control as far as possible. This might be done, for example, by supplementing the BCI with a single switch device which is meant to be used very occasionally. Failing this, the end user should be given a choice as to how they wish to address the issue. It should be noted that for the participant L01 who participated in the BCI study, his level of BCI control was far better than in the BCI Simulator experienced by healthy users, where the control settings were intentionally set for the system to unlock every 10 or 15 minutes. Thus, it is possible that the locking and unlocking function implemented in the current prototype would be sufficient for some BCI users.

Preference for playlist selection methods. The second purpose of the user studies was to establish how participants would wish to select their music with an increasing degradation of control. Playlist selection methods were selected where there would be a trade-off between ease (speed) of selection and precision of music. It was expected that with decreasing control, participants' choice of music selection would become progressively less precise in order to select music more quickly.

The first finding is that there is individual variation in users' baseline preference for imprecision in music selection. The way in which music is selected, and the way this changes as control degrades, depends on a variety of factors such as personal preference, habits, current context, mood, availability of other options, desire to listen to music, and the tolerance for imprecision and uncertainty in control. In particular, the context appeared to affect how participants chose to select music: in situations where specific [types of] music needed to be played, behaviours emerge where either the tolerance for playing music in degraded control circumstances was low or tolerance for selecting a specific track was high.

In all user groups, participants who were interested in or enthusiastic about the mood playing functionality were observed. The moods functionality itself is interesting since it can be tailored either for simplicity or complexity. Some participants appreciated the simplicity of selections, while others such as Brenda expressed a desire to invest time in fine-tuning the mood parameters. She was also enthusiastic about the principle of fine-tuning the parameters in order to be able to select a mood quickly in the music player. However, the eventual uptake of the mood (intelligent playlist generation) playlist selector depended on whether people thought the system filtered songs according to their mental model of the selected mood or genre. There were those who did not like the mood functionality but were willing to accept it as a 'best resort' option in situations where there would be a decrease in communication rate, while others would simply prefer to randomly shuffle their music. In the longitudinal study, there were also two participants for whom the mood selector seemed

to grow on with actual usage.

Again, in all studies, subsets of participants expressed a desire to be able to select individual tracks or albums, at least in some contexts. Some of the strongest reactions arose with regard to being able to select specific songs to play: three participants in the video prototyping study expressed strong opinions of wanting to select precise songs, the composer in the prototype study wanted to be able to scan through tracks, and the one BCI participant who had a major disability was very happy to be able to select an album of his choosing. These reactions were somewhat confirmed in the studies with the online simulator where, with diminishing levels of control, there were people who would still use the album selection; conversely, others stopped using the music player because they were not able to easily select specific songs. Thus, the results from the online simulations somewhat strengthened the findings from disabled users that album and song selection are an important goal for listeners of music, which may or may not be replaceable with other less precise methods of playing music depending on a given user.

A logical conclusion is that a music enthusiast who finds him or herself in a state of severe disability might well wish to invest time in selecting a particular song, and even to do other more complex and time-consuming tasks such as create individual playlists. This makes sense if one considers that regaining a feeling of control provides a sense of empowerment to the user, and is one of the factors in a person's choice to adopt an assistive technology (Pape et al., 2002). In the long run, it is possible that a given user would switch to intelligent playlists or random shuffle, and thus it is beneficial to develop and provide these options. However, as it was shown that end users are likely at least initially to be able to choose their own desired songs, it is recommended that the album and track selection mechanism is further developed. Customisation based on context may also be desirable, and algorithms or artificial intelligence techniques to enable users to quickly and easily identify their desired music. Ways of allowing users to easily explore or browse a music collection should also be explored, as disabled participants also mentioned the desire to be able to scan through albums and tracks linearly; it is recommended that as much as possible, they enable a person to feel in control of the browsing.

Functionality. In general, participants appreciated the simplicity of the music player prototype. Ease of use was repeatedly identified by participants as the reason why they liked or did not like their current solutions, and people stopped using the music player when it became difficult to use. Two functions that were identified by different groups in the prototype study as being missing were volume control and repeat. Thus, further developments of a minimalistic music player should at least seek to provide these options. Additional features such as creating playlists and having a stored history are also potential areas of future exploration. To minimise unnecessary cost to the user of having extraneous functions which they might never use, the features can be customised for each individual. Customisation is a usual part of applying an assistive solution (Sutcliffe et al., 2003).

6.6 General Discussion

The UCD process described in Chapter 2 was used to design a music player intended for use with a BCI by persons with LiS. In this section, reflections on several aspects of the process in general, and of the use of online simulations, is discussed.

Reflections on the use of video prototyping and early engagement with disabled users. User involvement was employed in four stages: a user requirements capture questionnaire, video prototyping, prototype evaluation without BCI, and prototype evaluation with BCI. Engaging with target end users at the beginning of the design process, as early on as possible, has been shown to greatly benefit the final product as the findings can uncover previously unthought-of needs and perspectives, substantially influencing the direction of the design (e.g. Buxton (2007)). This is particularly true for developing for persons with disabilities as the gap between the designer and end user's perspective is larger still (Dong et al., 2005). The current work attempted to add to this body of literature by eliciting feedback from disabled participants who may not have the level of physical disability as the intended target user, in a design process for an application using an unconventional input which they have not had previous exposure to.

Low-fidelity video prototypes were used to simulate potential behaviours of a BCI application, and participants were asked to provide opinions on how they might wish to use such a system. This engaged participants quickly at a low cost to all parties involved, and provided access to users' initial reactions to a novel interactive system without having to involve them in rigorous and potentially tiring usability testing. Although participants did not actively engage (interact) with the videos, simply demonstrating the limitations of BCI provided valuable insights into end users' expectations and requirements. The results led to steering the direction and focus of the music player towards empowering a user to precisely select their choice of music rather than on other potentially worthwhile goals such as engaging users in social interaction through music. The exercise may be considered a success, as it was demonstrated that the final prototype was well-received by severely disabled participants in later stages of development.

The design process sought to involve a larger number of disabled participants at the beginning of the study and at the end where evaluation of the final prototype could be assessed. This aimed to reduce the costs of involving disabled participants in long and tedious tests to establish the usability of the system. The implicit reasoning was that disabled end users' perspectives, needs and requirements would be somewhat different from that of non-disabled people. As such, the user studies were not set up in such a way that participant preferences could be compared, and in any case the sample sizes in each group were too small to make any statistical comparisons. Thus, one cannot really conclude that different conclusions would be drawn from having able-bodied users carrying out the video prototyping exercise rather than disabled end users. On the other hand, the choice of preference for the three playlist selection options in all studies with able-bodied users was skewed towards random

or moods in the BCI [simulator] modes, and none of the participants were overly thrilled with the album selection (although one did mention that he liked the binary selection tree). Conversely, the opinions of the most severely disabled participants leaned strongly towards the desire for control at the expense of time taken. It is therefore not unreasonable to speculate that if only healthy participants had been consulted at the beginning of the study, that the desire for precise selection of music would not have been picked up as clearly. This lends credence to the methodology of consulting with disabled users at the beginning of the design process.

Reflections on the use of the online BCI simulator for usability testing. The intention of using the online BCI simulator was to evaluate a BCI application without having to incur the costs of actually using a BCI. This allowed for involving a larger number of users in a shorter period of time for usability testing, and to deploy the system in participants' own contexts in the longitudinal study. The value of such an activity was two-fold: firstly, the main usability issues with regard to the application, such as some users' difficulties in selecting albums, and the main sources of frustration, could be uncovered using the online simulator. Secondly, both qualitative and quantitative changes in use of the application were observed when the level of control degraded past a user-specified threshold of tolerance. This enables the prediction of the range of behaviours that might occur in a real application, despite the limitations of only carrying out BCI studies in the lab. However, validation of user studies using the online simulator is required to find out to what extent the simulator is an adequate representation of actual BCI.

Going by participants' usage of the music player in BCI Simulator mode during the longitudinal study and stated preferences in the BCI study, there was a slight difference in the distribution of user preferences for the playlist methods: 2, 3 and 1 participants preferred the random, moods and album selectors for the real BCI mode respectively, and 4 and 2 participants preferred the random and mood functions in the BCI simulator mode. For the small sample sizes, the results are certainly comparable, and the conclusion that user preferences in the BCI simulator mode were more skewed towards random than the BCI mode is not possible to make. However, as people who participated in real BCI trials indicated, there was some inclination to believe that participants' performance in real BCI at least felt, if not was, subjectively better than for the real BCI. In this case, the skewed distribution of the simulator preferences may reflect the hypothesis that diminished control correlates with a preference for less precision. On the other hand, the discrepancy could have occurred because of the valid experience of the control characteristics in the longitudinal study, which would not have been evident with a real BCI. Thus, this highlights the importance of building simulators that adequately represent the true control characteristics of the BCI.

Reflections on individual differences. In a paper evaluating long term user experience, Kujala et al. (2011) states that 'User experience is personal–different users had diverging reactions even to the same [product]'. In the case of music listening using a MI-BCI,

users showed similarities and differences in their choices; however the reasoning behind their choices was sometimes not the same. For example, enjoyment could be due to the BCI interface being 'challenging, like a game', or that 'I wanted to explore the music collection'. The origin of individual differences common to both disabled and non-disabled people includes musical preferences and prior expectations, context of music use and openness to new technology or experiences. User tolerance to control adds another dimension to the list. Apart from one participant, all the participants in the longitudinal BCI simulator study (6 out of 6) and the BCI study (5 out of 6) preferred random or mood as their method of choice. Conversely, two out of four of the disabled participants in the BCI study would prefer the album selection, one would use a combination and another would have chosen the moods functionality.

It is worth emphasising that prior to the video prototyping study, it had been expected that disabled participants would be rather more interested in the intelligent smart playlist functionality. Although some users were very enthusiastic about the technology, the strong sense of a desire for achieving precision in music selection was surprising. However, it is known that a sense of control is important for people who have acquired physical disabilities, and that this influences their acceptance of [assistive] technology along with a myriad other personal factors (Pape et al. (2002), Scherer et al. (2005)). On reflecting that the appeal of randomly shuffling one's music is the surprise, unpredictability, or serendipity of not knowing exactly what is coming up next (Quiones, 2007), light is shed on a reason as to why someone with a physical disability might prefer to exert precise control at the cost of extra time: the thought of voluntarily relinquishing control is unappealing. The suggestion participants gave of scanning through one's music linearly is an extension of this: although browsing in this way may take a long time and enables one to find unexpected songs (Cunningham et al., 2004), this is different from simply randomising the music as here, the user is in control of the actual scanning.

Relatedly, people with severe disabilities are not constrained by the need to interact with a machine quickly with a given input modality (Birbaumer, 2006), but define their own sense of acceptable time to achieve tasks (Pape et al., 2002). It is likely that able-bodied participants' tolerance for a low communication rate and motivation to exert control cannot match a severely disabled user, and thus the threshold at which an able-bodied user either gives up playing music or gives up precision due to loss of input control is lower than that of an end user. One implication of this is that research into how to browse, interact with and navigate a large music collection easily with BCI control is a promising direction for future work: because the onset of fatigue can be fast, it makes sense to support the user in achieving their goals by increasing their rate of communication as far as possible even though they may not inherently mind exerting more effort.

Another factor which may influence preference is the desire for self-identification: a 'preserved self-image' (Pape et al., 2002). For example, less severely disabled people might view the simplification of music selection as being patronising or 'dumbing down' the interface,

and might reject the interface for that reason. BCIs studies indicate that the level of motivation to use a BCI can be significantly higher in participants with a significant level of physical disability than in people with less severe disabilities (Nijboer et al., 2010). Combining the differences in inherent preferences for control, musical tastes and physical ability means that for people with LiS, the range of preferences is likely to be as diverse as was experienced in the sample of healthy and disabled users.

Reflections on the overall process. It should be noted that the process of design is not linear, but involves iteratively developing and testing the system. Within this process, much of the iteration was done with healthy participants using both the simulator and real BCI, and with the author/designer trying out different configurations and exploring the system. For example, an early prototype used a scanning system which never reached the real BCI testing stage because through use of the simulator it was found not to provide a very good user experience. The final prototype described in this chapter can be thought to be a version which incorporates much of the functionality that people would desire, ripe for being fed into the next iteration of development where features are refined and other important ones added in.

To summarise, each user study had its own benefits and limitations. The requirements questionnaire allowed the identification of the expressed needs and goals of participants, but not to assess how participants might choose to control the system under BCI constraints. The video prototypes engaged disabled users for this purpose, but the implications for design could not be taken to be conclusive as it did not provide a means of allowing them to interact with the system. The question as to what users might actually choose to do in situations of reduced input control could be answered through the lab and longitudinal user studies with able-bodied users, but this user group was plainly the furthest away from the target user group. Again, asking disabled participants to evaluate the prototype using input modalities they were comfortable with was valuable: it allowed us to see what was important to users and to assess the music player as an actual application. However, the limitations were that most of the evaluations were only carried out in the lab, allowing only brief feedback, that participants with less severe disabilities might not appreciate the need for slower inputs, and that the limitations of BCI control could not be conveyed easily. Finally, evaluations with BCI, while clearly being the most accessible and relevant to the target user group, were the most costly and did not allow for a long term evaluation. We were also not able to test the system with an actual person with LiS.

Thus, by integrating the different perspectives afforded by the mixed-users, mixed-methods approach, it was possible to establish the *range* of preferences and behaviours that might be observed for a given target end user. This allowed the synthesis of a clear set of design recommendations for development of an application that one can be fairly confident would appeal to an LiS end user with a prior history of listening to music.

6.7 Conclusions

This chapter demonstrated the use of online simulations to design and develop a MI-BCI controlled music player intended for use by a person with LiS. Video prototypes were presented to disabled participants to find out how they would wish to control and select music, and non-disabled participants used the BCI Simulator from Chapter 4 to control a prototype in a lab session and a longitudinal session. Results from the simulation studies were combined with those from the initial interviews presented to disabled participants, prototype studies with disabled and able-bodied participants using BCI and other input modalities. Since there was a small number of participants in each group, statistical comparisons could not be made between groups but observed behaviours and subjective feedback from participants could be identified and used to develop guidelines for design.

The studies provide the first set of evidence demonstrating the value of using online simulations to reduce the cost of BCI testing with end users. The video prototype study set out to explore how participants would choose to select music with a degraded level of control as presented with a BCI. Although it was expected that participants would be excited about exploring new functionality with automatic playlists, and that this would be well received if the potentially poor control of BCI was demonstrated, there were those who expressly stated that it was important for them to be able to choose their own songs to play. That there is much value in this was shown when a participant with a major disability expressed delight in being able to select an album of his choosing using BCI control.

Although there can be no replacement for end user input especially while gathering the initial requirements and evaluation of a final prototype, this chapter also showed that the identification of major usability issues and insight into individual differences can be gained through online simulation studies with able-bodied users. In particular, the longitudinal studies allowed the break down of what aspects of control are particularly important or frustrating, which would have been difficult and costly to obtain with real BCI. The combination of results from a variety of personalities and level of physical ability naturally lead to sound conclusions about the most important features and functions that should be considered in the final design. Customisability is paramount for design: for many design decisions there is no one-size-fits-all, but the best solution is one that provides the best experience for an individual user (Scherer et al., 2005).

7 Conclusions and Future Outlook

7.1 Introduction

Brain-Computer Interfaces (BCIs) have the potential to enhance the quality of life of people living with Locked-in Syndrome (LiS) by improving their ability to communicate and have control over their environment. Until recently, the field of BCI research has been dominated by the technical concerns of achieving an acceptable rate of communication with BCIs. In transferring systems from the laboratory to end users, a number of challenges arise: designers will likely not have easy access to individuals, there are a myriad of individual differences which means systems must be customised for each individual, and carrying out user studies with BCI is costly. This thesis has presented a range of simulation techniques and tools which seek to reduce costs and speed up design and development of BCI applications.

Although some simulation tools and techniques have been used in BCI research, a discussion on the various techniques that are available, and their potential use in design and development, has not been published. This thesis has thus aimed to identify and consolidate potential techniques which can be used. In chapter 2 a user-centred design (UCD) process was described, showing the relevant parts of the process where different techniques can potentially be used to reduce costs or otherwise contribute to design and development. The work focuses specifically on BCIs driven by the motor-imagery (MI) paradigm. Two threads that have run through the thesis are offline and online simulations, and the findings from these are discussed separately.

7.2 Summary of research contributions

7.2.1 Offline simulation

Use of task times to predict usability and performance. BCI performance is typically reported in terms of abstract measures such as the theoretical bit rate or the information transfer rate. In Chapter 3, task completion time was introduced as a metric for comparing user interfaces in simulation studies. In particular, it was shown that a lower expected time does not necessarily correspond to a narrower prediction interval. This may be important as users may be more prone to remembering the longest task times, which could negatively affect the user experience of an interface which has a longer tail in the distribution of task times yet while having lower expected times. Simulation is thus shown to be a useful tool

for estimating task performance.

Trade-off between speed and accuracy in a binary selection task. Chapter 3 also investigated the potential benefit of a speed-accuracy trade off in a binary selection task. The effect of selection accuracy (proportion of correct trials) and time-to-selection (time to make one binary decision, TTS) on overall task time was explored through a simulation study. It was shown that the overall time taken to achieve a task for a low selection accuracy can be made comparable to, or even improve on, a higher selection accuracy by decreasing the TTS. For example, a task can be achieved more quickly for a user model with a selection accuracy of 0.7 and 1s TTS than a model with a selection accuracy of 0.8 and 4s TTS. One implication is that, if by increasing the TTS to accumulate evidence of the user's intent, only a small improvement on the selection accuracy can be made, it might not benefit the overall task performance.

Development of simulation models. The choice of abstraction level of a model is influenced by the purpose of the simulation. Two levels of modelling were demonstrated in this thesis: firstly, a simple user model represented by selection accuracy and time-to-selection was used for discrete binary simulations in Chapter 3. Simulating task performance using a combination of the two parameters as input into a finite state machine representation of an interface has not previously been investigated in the BCI literature. Chapter 4 described the development of a simulator which models the low-level characteristics of BCI. This was the first attempt in the literature to simulate the classifier output of a MI-BCI. Two methods of simulating the classifier output were described with promising results, and it was shown that a data-driven model based on modelling the frequency content of the signals provided a slightly better fit to the data than a generative Markov Chain model.

Use of offline simulation to investigate a novel selection mechanism. Chapter 5 reports a simulation experiment that simulates the performance of individual users in a novel selection mechanism using data from standard binary calibration trials. It was shown that different models can give rise to a range of possible user behaviours and performances, which could be useful for an initial analysis of a novel paradigm. In controlling the REx selection mechanism, a user's selection accuracy may not be uniform across targets in the wheel. It was found that to a large extent, the individual models simulated a range of behaviours such that collectively they were able to capture the selection accuracy for each target. However, it may be necessary to extend the initial calibration data collected, if it is even possible to make accurate predictions from the simulation models.

Comparison of simulation predictions with real data. Although numerical and analytical predictions of user performance based on the theoretical selection accuracy have been described in the literature, there have been no publications comparing simulated and actual task performance of a MI-BCI for individual users. In Chapter 3, simulations of task times from a previous session were used to provide predictions of time-to-task for given participants. The results for six participants indicate that this is feasible, as the expected value of the time-to-task and number of selections was within an acceptable range, and the metrics

fell within the simulation prediction intervals. In Chapter 5, it was shown that ensemble modelling or averaging of the predictions from different models better predicts the overall selection accuracy and time-to-selection of single trials in a novel selection mechanism.

7.2.2 Online Simulation

Development and evaluation of an online BCI simulator. The low-level simulator in Chapter 4 was intended to simulate the control characteristics of a MI-BCI. Different validation techniques that could be used to evaluate the simulator were discussed. An evaluation technique, the Turing test, was applied to provide preliminary validation of the simulator. Preliminary evidence was found that participants could distinguish between a 'poor' simulator and real BCI, but not real BCI and a 'good', generic simulator.

Use of video prototyping to engage disabled end users in application design. Again, the demonstration of the use of video prototyping to simulate the control of a MI-BCI is a novel contribution to BCI literature. Participants were shown different options they would choose under situations of uncertainty in control of a music player, and an attempt was made to uncover their underlying requirements. This was used to drive the subsequent design of the application forward, and evaluations with disabled participants at the BCI evaluation stage proved the design decisions to be a step in the right direction.

Use of the online simulator for development and debugging. The online simulator was used extensively in the development of the application and selection mechanism described in Chapters 5 and 6. Because the simulator's outputs are the same as that of a real BCI, it was possible to uncover usability and system issues that would not otherwise have been found until the real BCI trials. This has also been demonstrated in Boland et al. (2011), in the context of a chess application controlled with a visual ERP-based BCI. Although not all the issues can be uncovered with a simulator as there will be BCI-specific considerations, such as how difficult it is to actually produce the mental states and how this affects control of the BCI, some usability issues with regard to control are experienced. Interestingly, it was also possible to identify a potential control strategy for actual BCI using the BCI simulator, although the usefulness of this in real BCI control needs to be validated.

Use of the online simulator as an experience prototype. Chapter 6 also used the online simulator to elicit able-bodied participants' responses to simulated disability. It was shown that the uncertainty in control of the simulator induced frustration beyond simply having a slower input as with an 'intermediate' simulator. User tolerance was seen to differ between participants: one commented that she wouldn't use the BCI control while another said that she would work to use it if she had to. A longitudinal study also showed that participants vary in their behavioural responses to degradation of control in the context of playing music, and enabled the identification of the main sources of frustration with the application that arose from the lack of control. However, it is not clear that these

experiences would translate to a LiS user using an actual BCI to control their music.

7.2.3 Other contributions

Development of a novel BCI selection mechanism. Although the Hex-O-Spell developed by Williamson et al. (2009) is widely cited in BCI literature as a successful application for BCI, no studies on people's ability to use the actual selection mechanism have thus far been reported in the literature. The work in this thesis showed that a generalisation of the Hex-O-Spell, the Rotate-Extend introduced in Chapters 5 and 6, could be used by both able-bodied and disabled participants. The selection mechanism has the advantage that the user can select one out of several items in a single trial, and that it potentially creates a better user experience as the user has more control over the pace of interaction compared to a binary paradigm. User performance for the REx mechanism is best if there is a user bias such that the classifier output naturally leans towards one MI class. On the other hand, it may be difficult for a person with good binary performance to use the system, such as was shown for two users in the experiment in Chapter 5.

Potential link between disabled users' desire for control and preference for music selection. Although there is no way to infer a statistical difference in the music selection preferences of people with and without physical disabilities from the study in Chapter 6, qualitative analysis of interviews with disabled end users suggests that there may be an increased preference for being able to select music of one's own choosing, rather than randomly shuffling a music collection such as is popular with the general population (although long-term behaviour may be different from users' statements). This may be surprising as the level of effort required for a person with a physical disability to interact with the system is higher, and thus more effort might be saved in having the system select songs.

One possibility is that, at least for people with an acquired physical disability, regaining a sense of control is important. This would be in line with the literature pointing out that the motivation of physically disabled persons to interact with new technology and their drive to achieve desired goals without regard for effort and time-cost should not be underestimated. Further research into the way users with and without disabilities listen to music may be important in designing music systems for both user groups.

Insights into a UCD process for development of applications for severely disabled end users. Chapter 6 documents an iteration of a UCD process where disabled end users have been involved from user requirements capture, to implementation, to evaluation stages, has been described for a BCI application intended for an LiS end user. Although an actual user with LiS did not evaluate the final prototype at the end, the process can be considered a success as a person with tetraplegia was able to use and enjoyed using the BCI music player. However, it is not possible to make conclusions about the long-term use of the application. Limitations of the current BCI technology to be used by the end user also cannot be ignored. It was shown that the mixed-users, mixed-methods approach

strengthened the recommendations for designing the system, and thus the process can be recommended as an approach to designing future applications for BCI.

7.3 Future Directions for Research

The work described in this thesis has raised many research questions deserving further investigation. This section describes a select set of avenues for future work.

Validation of the use of simulations in the UCD process. In this research, online simulation techniques were shown to be useful in the UCD process. One direction for future work would be to further investigate the benefits and limitations of specific simulation techniques for specific purposes, and the types of questions that different tools are best at answering. One might also wish to evaluate how well the conclusions from simulation studies translate to longitudinal use of an actual application.

Investigating feedback and control. The simulator aims to isolate control characteristics of a real BCI. An interesting area of research would be to find out how these isolated features affect the user's ability to control a BCI, and what phenomena arise only from having a human brain in the loop. For example, the level of noise in visual feedback (e.g. noisy cursor movements) may affect a user's ability to control the feedback because of an extra cognitive load; on the other hand it may give rise to EEG artifacts that affect the features extracted and are out of the user's control. The simulator could be used to help answer these and other such questions.

Development and improvement of user models. Further knowledge of BCI control can be used to improve the user models. For example, the influence of feedback could be investigated in order to improve the simulator such that it these effects into account. This could then be used to provide better predictions of user performance for actual tasks.

Testing and investigating other selection mechanisms. Two different selection mechanisms were investigated in this thesis: the binary selection paradigm and the Rotate-Extend (REx) controller. The simulator could also similarly be used to investigate other selection mechanisms without having to use a real BCI. This would allow one to weed out selection mechanisms that are unlikely to work.

Extension to other BCI paradigms. The simulation tools described in this research are designed for motor-imagery and other mental-state based BCIs, and can be easily extended to include systems with more than 2 classes. The use of simulation techniques to aid the design process can also be applied to other BCI systems such as stimulus-driven paradigms.

7.4 Conclusions

This thesis has investigated how simulation techniques can be used in the design and development process of BCI applications. Although it can take time to develop the models and prototypes, it has been shown that the benefits of using these tools can outweigh the costs to designers, developers and end users alike. User models can be used to run offline simulations which provide reliable estimates of task performance for a range of users, enabling one to select the designs that maximise task performance out of a set of options for a given user. Online simulations can realistically present the control characteristics of BCI, which may be used to engage stakeholders in understanding a BCI without having to train and use the system, and to explore selection mechanisms and tune application parameters. The techniques can be used to identify usability problems and design solutions in advance of carrying out actual BCI experiments, which are time-consuming and effortful for end users and even healthy participants as they typically involve many repeated tasks over several sessions. The wide range of inter- and intra-individual differences can also be taken into account without involving BCI users across the whole spectrum of performances. Thus, the substantial costs of modelling can significantly outweigh the costs and ethical considerations arising from travelling to visit an end user, setting up the BCI and carrying out repeated usability studies. Using and improving the tools and techniques described in this thesis can be expected to enhance the quality and speed of designing and developing BCI applications for people with LiS.

Bibliography

R. Adams, G.S. Bahr, and B. Moreno. Brain Computer Interfaces: Psychology and Pragmatic Perspectives for the Future. In *AISB 2008 Convention Communication, Interaction and Social Intelligence*, volume 1, page 1, 2008.

H. Al-Nashash, Y. Al-Assaf, J. Paul, and N. Thakor. EEG signal modeling using adaptive Markov process amplitude. *IEEE transactions on Biomedical Engineering*, 51(5):744–751, 2004.

Aricò P. Schettini-F. Riccio A. Salinari S. Mattia D. Babiloni F. Cincotti F. Aloise, F. A covert attention p300-based brain computer interface: Geospell. *Ergonomics*, 55, 2012.

F. Aloisea, I. Lasorsaa, F. Schettini, A.M Brouwerd, D. Mattiaa, F. Babilonia, S. Salinaric, MG Marciania, and F. Cincottia. Multimodal stimulation for a P300-based BCI. *International Journal of Bioelectromagnetism*, 9(3):128–130, 2007.

B. Arslan, A. Brouse, J. Castet, J.J. Filatriau, R. Lehembre, Q. Noirhomme, and C. Simon. Biologically-driven musical instrument. In *eINTERFACE'05-Summer Workshop on Multimodal Interfaces*, 2005.

S. Ball and J. Rousell. Virtual Disability: Simulations as an Aid to Lecturers' Understanding of Disability. *Computers Helping People with Special Needs*, pages 624–624, 2004.

A. Bangor, P.T. Kortum, and J.T. Miller. An Empirical Evaluation of the System Usability Scale. *International Journal of Human-Computer Interaction*, 24(6):574–594, 2008.

J. Banks, B.L. Nelson, and D.M. Nicol. *Discrete-event system simulation*. Prentice Hall, 2009.

A. Bashashati, R.K. Ward, and G.E. Birch. Towards development of a 3-state self-paced brain-computer interface. *Computational Intelligence and Neuroscience*, 2007.

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

M.F. Bear, B.W. Connors, and M.A. Paradiso. *Neuroscience: Exploring the Brain*. Lippincott Williams and Wilkins, 3rd edition, 2006.

J. Benedek and T. Miner. Measuring desirability: New methods for evaluating desirability in a usability lab setting. In *Usability Professionals' Association*, 2002.

M. Bensch, A.A. Karim, J. Mellinger, T. Hinterberger, M. Tangermann, M. Bogdan, W. Rosenstiel, and N. Birbaumer. Nessi: an EEG-controlled web browser for severely

paralyzed patients. *Computational Intelligence and Neuroscience*, 2007.

N.O. Bernsen, H. Dybkjær, and L. Dybkjær. Wizard of Oz prototyping: How and when. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '93, 1993.

M. Betke. Intelligent interfaces to empower people with disabilities. In H. Nakashima, H. Aghajan, and J.C. Augusto, editors, *Handbook of Ambient Intelligence and Smart Environments*, pages 409–432. Springer, 2010.

F. Beverina, G. Palmas, S. Silvoni, F. Piccione, and S. Giove. User adaptive BCIs: SSVEP and P300 based interfaces. *PsychNology Journal*, 1(4):331–354, 2003.

S. Bhattacharya, A. Basu, and D. Samanta. Computational modeling of user errors for the design of virtual scanning keyboards. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(4), 2008.

L. Bianchi, L.R. Quitadamo, G. Garreffa, G.C. Cardarilli, and M.G. Marciani. Performances evaluation and optimization of brain computer interface systems in a copy spelling task. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(2):207–216, 2007.

M. Billinger, I. Daly, V. Kaiser, J. Jin, B.Z. Allison, G.R. Müller-Putz, and C. Brunner. Is it significant? guidelines for reporting bci performance. In Brendan Z. Allison, Stephen Dunne, Robert Leeb, José Del R. Millán, and Anton Nijholt, editors, *Towards Practical Brain-Computer Interfaces*, Biological and Medical Physics, Biomedical Engineering, pages 333–354. Springer Berlin Heidelberg, 2013.

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

N. Birbaumer and L.G. Cohen. Brain–computer interfaces: communication and restoration of movement in paralysis. *The Journal of Physiology*, 579(3):621, 2007.

N. Birbaumer, T. Elbert, B. Rockstroh, and W. Lutzenberger. Biofeedback of event-related slow potentials of the brain. *International Journal of Psychology*, 16(4):389–415, 1981.

C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

P. Biswas and P. Langdon. Towards an inclusive world - a simulation tool to design interactive electronic systems for elderly and disabled users. In *Annual SRII Global Conference*, 2011.

P. Biswas and P. Robinson. Simulation to predict performance of assistive interfaces. In

Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility, page 228. ACM, 2007.

P. Biswas and P. Robinson. A new screen scanning system based on clustering screen objects. *Journal of Assistive Technologies*, 2(3):24–31, 2008a.

P. Biswas and P. Robinson. Automatic evaluation of assistive interfaces. In *Proceedings of the 13th international conference on Intelligent user interfaces*, pages 247–256. ACM, 2008b.

B. Blankertz and C. Vidaurre. Towards a cure for BCI illiteracy: machine learning based co-adaptive learning. *BMC Neuroscience*, 10(Suppl 1):P85, 2009.

B. Blankertz, G. Dornhege, M. Krauledat, K. Müller, V. Kunzmann, F. Losch, and G. Curio. The berlin brain-computer interface: Eeg-based communication without subject training. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2006a.

B. Blankertz, G. Dornhege, M. Krauledat, M. Schröder, J. Williamson, R. Murray-Smith, and K.R. Müller. The Berlin brain-computer interface presents the novel mental typewriter hex-o-spell. In *Proceedings of the 3rd International Brain-Computer Interface Workshop and Training Course*, pages 108–109. Citeseer, 2006b.

B. Blankertz, F. Losch, M. Krauledat, G. Dornhege, G. Curio, and K.R. Müller. The berlin brain-computer interface: Accurate performance from first-session in bci-naïve subjects. *IEEE Transactions on Biomedical Engineering*, 55(10):2452–2462, 2008.

B. Blankertz, C. Sannelli, S. Halder, E.M. Hammer, A. Kübler, K.R. Müller, G. Curio, and T. Dickhaus. Neurophysiological predictor of SMR-based BCI performance. *NeuroImage*, 51(4):1303–1309, 2010. ISSN 1053-8119.

Dornhege G. Krauledat-M. Müller K.R. Curio G. Blankertz, B. The non-invasive berlin brain-computer interface: Fast acquisition of effective performance in untrained subjects. *NeuroImage*, 37(2):539–550, 2007.

D. Boland, M. Quek, M. Tangermann, R. Murray-Smith, and J. Williamson. Using simulated input into brain-computer interfaces for user-centred design. *International Journal of Bioelectromagnetism*, 13(2):86–87, 2011.

J.F. Borisoff, S.G. Mason, and G.E. Birch. Brain interface research for asynchronous control applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14 (2):160–164, 2006.

M. Buchenau and J.F. Suri. Experience prototyping. In *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques*, DIS '00,

pages 424–433, New York, NY, USA, 2000. ACM.

S. Burgstahler and T. Doe. Disability-related simulations: If, when, and how to use them in professional development. *Review of Disability Studies*, 1(2):4–17, 2004.

B. Buxton. Experience design vs. interface design. *Rotman Magazine, Winter*, pages 47–49, 2005.

B. Buxton. *Sketching User Experiences: Getting the Design Right and the Right Design.* Morgan Kaufmann, 2007.

Moran T.P. Newell-A. Card, S.K. The keystroke-level model for user performance with interactive systems. *Communications of the ACM*, 23:396–410, 1980.

Moran T.P. Newell-A. Card, S.K. *The Model Human Processor: An Engineering Model of Human Performance*, volume Handbook of Perception and Human Performance. Vol. 2: Cognitive Processes and Performance, pages 1–35. John Wiley & Sons, 1986.

L. Carroll. *Alice's Adventures in Wonderland.* Macmillan, 1865.

A. Chatterjee, V. Aggarwal, A. Ramos, S. Acharya, and N.V. Thakor. A brain-computer interface with vibrotactile biofeedback for haptic information. *Journal of NeuroEngineering and Rehabilitation*, 4(1):40, 2007.

F. Cincotti, L. Kauhanen, F. Aloise, T. Palomaki, N. Caporusso, P. Jylanki, D. Mattia, F. Babiloni, G. Vanacker, M. Nuttin, M. G. Marciani, and J. del R. Millán. Preliminary experimentation on vibrotactile feedback in the context of mu-rhythm based BCI. In *Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4739–4742, 2007a.

F. Cincotti, L. Kauhanen, F. Aloise, T. Palomaki, N. Caporusso, P. Jylanki, D. Mattia, F. Babiloni, G. Vanacker, M. Nuttin, et al. Vibrotactile feedback for brain-computer interface operation. *Computational Intelligence and Neuroscience*, 2007:48937, 2007b.

J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(37):37–46, 1960.

B. Costello and E. Edmonds. A study in play, pleasure and interaction design. In *Proceedings of the 2007 conference on Designing pleasurable products and interfaces*, page 91. ACM, 2007.

Nuyujukian P. Gilja-V. Chestek C. Ryu S.I. Shenoy K.V. Cunningham, J.P. A closed-loop human simulator for investigating the role a closed-loop human simulator for investigating the role of feedback control in brain-machine interfaces. *Journal of Neurophysiology*, 105,

2010.

S.J. Cunningham, M. Jones, and S. Jones. Organizing digital music for use: An examination of personal music collections. In *Proceedings of The International Conference on Music Information Retrieval*, 2004.

A. Curran and M. J. Stokes. Learning to control brain activity: A review of the production and control of eeg components for driving brain-computer interface (bci) systems. *Brain and Cognition*, 51(3):326–336, 2003.

S. Hacker M. A. Bruno A. Demertzi F. Pellas S. Laureys A. Kubler D. Lule, C. Zickler. Life can be worth living in locked-in syndrome. *Progress in brain research*, 177:339–351, 2009.

Suganthan P.N. Das, S. Differential evolution: A survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 15(1), 2011.

R.J. Davidson. What does the prefrontal cortex "do" in affect: Perspectives on frontal EEG asymmetry research. *Biological Psychology*, 67(1-2):219–234, 2004.

R.C. Davis, T.S. Saponas, M. Shilman, and J.A. Landay. SketchWizard: Wizard of Oz prototyping of pen-based user interfaces. In *Proceedings of the 20th annual ACM symposium on User Interface Software and Technology*, UIST '07, pages 119–128. ACM, 2007.

P. Desain, J. Farquhar, P. Haselager, C. Hesse, and R. Schaefer. What BCI research needs. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '08, 2008.

P. Desmet and E. Dijkhuis. A wheelchair can be fun: a case of emotion-driven design. In *Proceedings of the 2003 International Conference on Designing Pleasurable Products and Interfaces*, pages 22–27. ACM New York, NY, USA, 2003.

E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematlk*, 1:269–271, 1959.

A. Dix. *HCI Models, Theories, and Frameworks*, chapter Upside down $\forall sandalgorithms --computationalformalismsandtheory.MorganKaufmann$, 2003.

A. Dix, J. Finlay, G.D. Abowd, and R. Beale. *Human-Computer Interaction.* Prentice hall, 2004.

J.E. Doble, A.J Haig, C. Anderson, and R. Katz. Impairment, activity, participation, life satisfaction, and survival in persons with locked-in syndrome for over a decade: Follow-up on a previously reported cohort. *Journal of Head Trauma Rehabililiation*, 18(5):435–444, 2003.

E. Doherty, G. Cockton, C. Bloor, and D. Benigno. Improving the performance of the cyberlink mental interface with "yes / no program". In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '01, pages 69–76, 2001.

E. Donchin, K.M. Spencer, and R. Wijesinghe. The mental prosthesis: Assessing the speed of a p300-based brain-computer interface. *IEEE Transactions on Rehabilitation Engineering*, 8:174–179, 2000.

H. Dong, P.J. Clarkson, J. Cassim, and S. Keates. Critical user forums – an effective user research method for inclusive design. *Design Journal*, 8(2):49–59, 2005.

G. Dornhege. *Increasing Information Transfer Rates for Brain-Computer Interfacing.* PhD thesis, University of Potsdam, Potsdam, Germany, 2006.

G. Dornhege, J.R. Millán, T. Hinterberger, D.J. McFarland, T.J. Sejnowski, and K.R. Muller, editors. *Toward Brain-Computer Interfacing.* The MIT Press, 2007.

Lucas J.P. Pisansky M.T. He B. Doud, A.J. Continuous three-dimensional control of a virtual helicopter using a motor imagery based brain-computer interface. *PLoS ONE*, 6 (10), 2011.

J.P. Ebert and D.M Wegner. Time warp: Authorship shapes the perceived timing of actions and events. *Consciousness and Cognition*, 19:481–489, 2010.

P. Eslambolchilar. *Making Sense of Interaction Using a Model-Based Approach.* PhD thesis, Hamilton Institute, National University of Ireland, Maynooth, 2006.

G. Evreinov and R. Raisamo. Optimizing menu selection process for single-switch manipulation. *Computers Helping People with Special Needs*, pages 628–628, 2004.

H.A. Fait and J. Mankoff. *EASE: A Simulation Tool for Accessible Design.* Computer Science Division, University of California, 2003.

S. Fazli, M. Danóczy, F. Popescu, B. Blankertz, and K.R. Müller. Using rest class and control paradigms for brain computer interfacing. In *Proceedings of the 10th International Work-Conference on Artificial Neural Networks: Part I: Bio-Inspired Systems: Computational and Ambient Intelligence*, IWANN '09, pages 651–665, 2009.

E.A. Felton, R.G. Radwin, J.A. Wilson, and J.C. Williams. Evaluation of a modified fitts law brain–computer interface target acquisition task in able and motor disabled individuals. *Journal of Neural Engineering*, 6(5), 2009.

P.M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, pages 381–391, 1954.

D. Friedman, R. Leeb, L. Dikovsky, M. Reiner, G. Pfurtrscheller, and M. Slater. Controlling a virtual body by thought in a highly-immersive virtual environment. In Vázquez P.P. Pereira J.M. Braz, J., editor, *Proceedings of the Second International Conference on Computer Graphics Theory and Applications*, pages 83–90. GRAPP '07, 2007.

D. Friedman, R. Leeb, G. Pfurtscheller, and M. Slater. Human-Computer Interaction Issues in Brain-Computer Interface and Virtual Reality. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, 2008.

E.V.C Friedrich, D.J. McFarland, C. Neuper, T.M. Vaughan, P. Brunner, and J.R. Wolpaw. A scanning protocol for a sensorimotor rhythm-based brain-computer interface. *Biological Psychology*, 80(2):169–175, 2009.

E.V.C Friedrich, R. Scherer, and C. Neuper. The effect of distinct mental strategies on classification performance for brain-computer interfaces. *International Journal of Psychophysiology*, 2012.

K.Z. Gajos. Understanding how to design complex brain-controlled applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, 2010.

F. Galán, M. Nuttin, E. Lew, PW Ferrez, G. Vanacker, J. Philips, and J.R. Millán. A brain-actuated wheelchair: Asynchronous and non-invasive Brain–computer interfaces for continuous control of robots. *Clinical Neurophysiology*, 119(9):2159–2169, 2008.

G. Garipelli, F. Galán, R. Chavarriaga, P.W. Ferrez, E. Lew, and J.R. Millán. The use of brain-computer interfacing for ambient intelligence. In *Constructing Ambient Intelligence*, AML '07. Springer, 2008.

N.A. Gates, C.K. Hauser, and E.W. Sellers. A longitudinal study of p300 brain-computer interface and progression of amyotrophic lateral sclerosis. In *Proceedings of the 6th international conference on Foundations of augmented cognition: directing the future of adaptive systems*, FAC '11, pages 475–483, 2011.

L. George and A. Lécuyer. An overview of research on "passive" brain-computer interfaces for implicit human-computer interaction. In *International Conference on Applied Bionics and Biomechanics ICABB 2010 - Workshop W1 "Brain-Computer Interfacing and Virtual Reality"*, 2010.

C.J. Gonsalvez and J. Polich. P300 amplitude is determined by target-to-target interval. *Psychophysiology*, 39(3):388–396, 2002.

M. Grosse-Wentrup and B. Schölkopf. A review of performance variations in smr-based braincomputer interfaces (bcis). In C. Guger, B.Z. Allison, and G. Edlinger, editors,

SpringerBriefs in Electrical and Computer Engineering. Springer, 2013.

C. Guger, A. Schlogl, C. Neuper, D. Walterspacher, T. Strein, and G. Pfurtscheller. Rapid prototyping of an eeg-based brain-computer interface (bci). *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 9(1):49–58, 2001.

C. Guger, G. Edlinger, W. Harkam, I. Niedermayer, and G. Pfurtrscheller. How many people are able to operate an eeg-based brain-computer interface (bci)? *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):145–147, 2003.

J. Hailpern, M. Danilevsky, A. Harris, K. Karahalios, G. Dell, and J. Hengst. Aces: promoting empathy towards aphasia through language distortion emulation software. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 609–618, New York, NY, USA, 2011. ACM.

S. Halder, A. Furdea, R. Leeb, G. Müller-Putz, A. Hösle, and A. Kübler. Implementation of smr based brain painting. In *Neuromath*, 2009.

Amento B. Kuznetsov S. Bell R. Harrison, C. Rethinking the progress bar. In *ACM Symposium on User Interface Software and Technology*, UIST '07, pages 115–118, 2007.

S.G. Hart and L.E. Staveland. *Arguing for Aesthetics in Human-Computer Interaction*, volume Human Mental Workload. North Holland Press, 1988.

M. Hassenzahl. The interplay of beauty, goodness, and usability in interactive products. *Human Computer Interaction*, 19(4):319–349, 2008.

M. Hassenzahl and N. Tractinsky. User experience–a research agenda. *Behaviour & Information Technology*, 25(2):91–97, 2006.

M. Hassenzahl and D. Ullrich. To do or not to do: Differences in user experience and retrospective judgments depending on the presence or absence of instrumental goals. *Interacting with computers*, 19(4):429–437, 2007.

M. Hassenzahl, R. Kekez, and M. Burmester. The importance of a software's pragmatic quality depends on usage modes. In *Proceedings of the 6th international conference on Work With Display Units*, WWDU '02, pages 275–276, 2002.

P. Hekkert, M. Mostert, and G. Stompff. Dancing with a machine: A case of experience-driven design. In *Proceedings of the 2003 international conference on Designing pleasurable products and interfaces*, pages 114–119. ACM New York, NY, USA, 2003.

Ganguly K. Jimenez J. Carmena J.M. Héliot, R. Learning in closed-loop brain–machine interfaces: Modeling and experimental validation. *IEEE Transactions on Systems, Man*

and Cybernetics – Part B: Vybernetics, 40(5), 2010.

T. Hermann and A. Hunt. An introduction to interactive sonification. *IEEE multimedia*, 12(2):20–24, 2005.

R. Herriot. *When Users Cannot be Included in Inclusive Design*, volume Designing Inclusive Systems. Springer, 2012.

W.E. Hick. On the rate of gain of information. *Quarterly Journal of Experimental Psychology*, 4:11–26, 1952.

T. Hinterberger, J. Hill, and N. Birbaumer. An auditory brain-computer communication device. In *Biomedical Circuits and Systems, 2004 IEEE International Workshop on*, pages S3/6–15–18, 2004a.

T. Hinterberger, N. Neumann, M. Pham, A. Kübler, A. Grether, N. Hofmayer, B. Wilhelm, H. Flor, and N. Birbaumer. A multimodal brain-based feedback and communication system. *Experimental brain research*, 154(4):521–526, 2004b.

T. Hinterberger, S. Schmidt, N. Neumann, J. Mellinger, B. Blankertz, G. Curio, and N. Birbaumer. Brain-computer communication and slow cortical potentials. *IEEE Transactions on Biomedical Engineering*, 51(6):1011–1018, 2004c.

D. Hitchcock and A. Taylor. Simulation for Inclusion–true user centred design. In *Proceedings of International Conference on Inclusive Design, Royal College of Art, London*, DIS '03. Citeseer, 2003.

D.R. Hitchcock, S. Lockyer, S. Cook, and C. Quigley. Third age usability and safety–an ergonomics contribution to design. *International Journal of Human-Computer Studies*, 55(4):635–643, 2001.

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(7099):164, 2006.

E. Hoggan, A. Crossan, S.A. Brewster, and T. Kaaresoja. Audio or tactile feedback: which modality when? In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, CHI '09, pages 2253–2256. ACM, 2009.

J. Höhne, M. Schreuder, B. Blankertz, and M. Tangermann. A novel 9-class auditory erp paradigm driving a predictive text entry system. *Frontiers in Neuroscience*, 5(99), 2011.

S. Houde and C. Hill. What do prototypes prototype? *Handbook of human-computer interaction*, 2:367–381, 1997.

H.J. Hwang, K. Kwon, and C.H. Im. Neurofeedback-based motor imagery training for brain–computer interface (BCI). *Journal of Neuroscience Methods*, 179(1):150–156, 2009.

R. Hyman. Stimulus information as a determinant of reaction time. *Journal of Experimental Psychology*, 45:188–196, 1953.

W. IJsselsteijn, W. van den Hoogen, C. Klimmt, Y. de Kort, C. Lindley, K. Mathiak, K. Poels, N. Ravaja, M. Turpeinen, and P. Vorderer. Measuring the experience of digital game enjoyment. *Proceedings of Measuring Behavior*, pages 88–89, 2008.

J. Jimenez, R. Heliot, and J.M. Carmena. Learning to use a brain-machine interface: Model, simulation and analysis. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, EMBC '09, pages 4551–4554, sept. 2009. doi: 10.1109/IEMBS.2009.5332718.

B.E. John. Using predictive human performance models to inspire and support ui design recommendations. In *SIGCHI Conference on Human Factors in Computing Systems*, 2011.

B.E. John and D.E. Kieras. Using goms for user interface design and evaluation: Which technique? *ACM Transactions on Computer-Human Interaction*, pages 287–319, 1996.

Prevas K. Salvucci D. Koedinger K. John, B. Predictive human performance modeling made easy. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, 2004.

J. Johnson. *Designing with the Mind in Mind: a Simple Guide to Understanding User Interface Design Rules*. Morgan Kaufmann, 2011.

W. Ju, B.A. Lee, and S.R. Klemmer. Range: exploring implicit interaction through electronic whiteboard design. In *Proceedings of the ACM 2008 conference on Computer Supported Cooperative Work*, pages 17–26. ACM, 2008.

S. Keates, J. Clarkson, and P. Robinson. Investigating the applicability of user models for motion-impaired users. In *Proceedings of the fourth international ACM conference on Assistive technologies*, pages 129–136. ACM, 2000.

R. Kerepki, B. Blankertz, G. Curio, and K.R. Müller. The berlin brain-computer interface (bbci)— towards a new communication channel for online control in gaming applications. *Multimedia Tools and Applications*, 33(1):73–90, 2007.

K. Khanna, A. Verma, and B. Richard. "the locked-in syndrome": Can it be unlocked? *Journal of Clinical Gerontology and Geriatrics*, 2(4):96–99, 2011.

D. Kieras. Model-based evaluation. In J.A. Jacko and S. Andrew, editors, *The human-computer interaction handbook*, chapter Model-based evaluation, pages 1139–1151. L. Erlbaum Associates Inc., 2003.

J.H. Kim, D.V. Gunn, E. Schuh, B. Phillips, R.J. Pagulayan, and D. Wixon. Tracking real-time user experience (true): a comprehensive instrumentation solution for complex systems. 2008.

S.C. Kleih, A. Riccio, D. Mattia, V. Kaiser, E.V.C. Friedrich, R. Scherer, G. Müller-Putz, C. Neuper, and A. Kübler. Motivation influences performance in smr-bci. In *5th International BCI Conference*, 2011.

D.S. Klobassa, T.M. Vaughan, P. Brunner, N.E Schwartz, J.R. Wolpaw, C. Neuper, and E.W. Sellers. Toward a high-throughput auditory P300-based brain–computer interface. *Clinical Neurophysiology*, 120(7):1252–1261, 2009.

M. Krauledat, M. Tangermann, B. Blankertz, and K.R. Müller. Towards zero training for brain-computer interfacing. *PLoS ONE*, 3(8), 2008.

J. Kronegg, G. Chanel, S. Voloshynovskiy, and T. Pun. Eeg-based synchronized brain-computer interfaces: A model for optimizing the number of mental tasks. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(1):50–58, 2007.

A. Kübler and K.R. Müller. *An introduction to brain-computer interfacing*, pages 1–25. Toward brain-computer interfacing. The MIT Press, 2007.

A. Kübler, B. Kotchoubey, J. Kaiser, J.P. Wolpaw, and N. Birbaumer. Brain-computer communication: unlocking the locked in. *Psychological Bulletin*, 127(3):358–375, 2001.

A. Kübler, N. Neumann, B. Wilhelm, T. Hinterberger, and N. Birbaumer. Predictability of brain–computer communication. *Journal of Psychophysiology*, 18:121–129, 2004.

S. Kujala, V. Roto, K. Väänänen-Vainio-Mattila, E. Karapanos, and A. Sinnelä. Ux curve: A method for evaluating long-term user experience. *Interacting with Computers*, 23(5): 473–483, 2011.

J. Kuljis. Hci and simulation packages. In *Proceedings of the 28th conference on Winter simulation*, WSC '96, pages 687–694, 1996.

S. Kullback and R.A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(49–86), 1951.

N. Lankton and D.H. McKnight. Using expectation disconfirmation theory to predict technology trust and usage continuance intentions. Technical report, Working Paper, Work-

shops of University of Minnesota, 2006.

S. Laureys, F. Pellas, P.V. Eeckhout, S. Ghorbel, C. Schnakers, F. Perrin, J. Berr, M.E. Faymonville, K.H. Pantke, F. Damas, M. Lamy, G. Moonen, and S. Goldman. The locked-in syndrome: what is it like to be conscious but paralyzed and voiceless? *Progress in brain research*, 150:495–512, 2005.

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

J.C. Lee and D.S. Tan. Using a low-cost electroencephalograph for task classification in HCI research. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*. ACM, 2006.

R. Leeb, C. Keinrath, D. Friedman, C. Guger, R. Scherer, C. Neuper, M. Garau, A. Antley, A. Steed, M. Slater, et al. Walking by thinking: the brainwaves are crucial, not the muscles! *Presence: Teleoperators and Virtual Environments*, 15(5):500–514, 2006.

R. Leeb, F. Lee, C. Keinrath, R. Scherer, H. Bischof, and G. Pfurtscheller. Brain-computer communication: Motivation, aim, and impact of exploring a virtual apartment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(4):473–482, 2007.

Y.K. Lim, E. Stolterman, and J. Tenenberg. The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 15(2):1–27, 2008.

N. Lofthouse, E. Arnold, S. Hersch, E. Hurt, and R. DeBeus. A review of neurofeedback treatment for pediatric adhd. *Journal of Attention Disorders*, 16(5):351–372, 2012.

F. Lotte. Generating artificial eeg signals to reduce bci calibration time. In *Proceedings of the 5th International Brain-Computer Interface Workshop*, 2011.

H. Lu, M. Wang, and H. Yu. EEG Model and Location in Brain when Enjoying Music. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 2695–2698, 2005.

M.T. Lynn, C.C. Berger, T.A. Riddle, and E. Morsella. Mind control? creating illusory intentions through a phony brain–computer interface. *Consciousness and Cognition*, 2010.

I.S. MacKenzie. *Motor Behaviour Models for Human-Computer Interaction*, pages 27–54. Morgan Kaufmann, 2003. URL `http://www.yorku.ca/mack/carroll.html`.

I.S. MacKenzie and R.W. Soukoreff. Phrase sets for evaluating text entry techniques. In *SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pages 754–755.

ACM, 2003.

S. Mahlke. Understanding users' experience of interaction. *Proceedings of the 2005 Annual Conference on European Association of Cognitive Ergonomics*, pages 251–254, 2005.

S. Mahlke. Studying user experience with digital audio players. *Lecture Notes in Computer Science*, 4161:358, 2006.

J. Mankoff, H. Fait, and R. Juang. Evaluating accessibility by simulating the experiences of users with vision or motor impairments. *IBM Systems Journal*, 44(3):505–517, 2010.

A. Maria. Introduction to modeling and simulation. In *Proceedings of the Winter Simulation Conference*, 1997.

S.G. Mason and G.E. Birch. A general framework for brain-computer interface design. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(1):70–85, 2003.

D.J. McFarland and J.R. Wolpaw. Eeg-based communication and control: Speed-accuracy relationships. *Applied Psychophysiology and Biofeedback*, 28:217–231, 2003.

D.J. McFarland, W.A. Sarnacki, and J.R. Wolpaw. Brain-computer interface (bci) operation: optimizing information transfer rates. *Biological Psychology*, 63(3):237 – 251, 2003.

D.J. McFarland, W.A. Sarnacki, and J.R Wolpaw. Electroencephalographic (eeg) control of three-dimensional movement. *Journal of Neural Engineering*, 7(3), 2010.

J.d.R. Millán, P.W. Ferrez, F. Galán, E. Lew, and R. Chavarriaga. Non-invasive brain-machine interaction. *International Journal of Pattern Recognition and Artificial Intelligence*, 22:959–972, 2008.

J.d.R. Millán, R. Rupp, G.R. Müller-Putz, R. Murray-Smith, C. Giugliemma, M. Tangermann, C. Vidaurre, F. Cincotti, A. K
"ubler, R. Leeb, C. Neuper, K.R. Müller, and D. Mattia. Combining brain-computer interfaces and assistive technologies: state-of-the-art and challenges. *Frontiers in Neuroscience*, 4:161, 2010.

J.R. Millán, F. Renkens, J. Mouriño, and W. Gerstner. Brain-actuated interaction. *Artificial Intelligence*, 159(1-2):241–259, 2004.

Mouriño J. Millán, J.R. Asynchronous bci and local neural classifiers: an overview of the adaptive brain interface project. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):159–161, 2003.

E.R. Miranda, S. Durrant, and T. Anders. Towards Brain-Computer Music Interfaces:

Progress and Challenges. In *Applied Sciences on Biomedical and Communication Technologies, 2008. ISABEL'08. First International Symposium on*, pages 1–5, 2008.

M. Mohri. Finite-state transducers in language and speech processing. *Computer Linguistics*, 23(2):269–311, 1997.

G.R. Müller-Putz, R. Scherer, C. Brunner, R. Leeb, and G. Pfurtscheller. Better than random? a closer look on bci results. *International Journal of Bioelectromagnetism*, 10 (1):52–55, 2008.

J.I. Münßinger, S. Halder, S.C. Kleih, A. Furdea, V. Raco, A. Hösle, and A. Kübler. Brain painting: first evaluation of a new brain–computer interface application with als-patients and healthy volunteers. *Frontiers in Neuroscience*, 4(182), 2010.

N. Neumann, A. Kübler, J. Kaiser, T. Hinterberger, and N. Birbaumer. Conscious perception of brain states: mental strategies for brain–computer communication. *Neuropsychologia*, 41(8):1028–1036, 2003.

C. Neuper and G. Pfurtscheller. Neurofeedback training for bci control. In B. Graimann, G. Pfurtscheller, and B.Z. Allison, editors, *Brain-Computer Interfaces*, The Frontiers Collection, pages 65–76. Springer, 2010.

J. Nielsen and R. Molich. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering people*, CHI '90, pages 249–256. ACM, 1990.

F. Nijboer, A. Furdea, I. Gunst, J. Mellinger, D.J. McFarland, N. Birbaumer, and A. Kübler. An auditory brain–computer interface (BCI). *Journal of Neuroscience Methods*, 167(1): 43–50, 2008a.

F. Nijboer, E.W Sellers, J. Mellinger, M.A Jordan, T. Matuz, A. Furdea, S. Halder, U. Mochty, D.J Krusienski, T.M Vaughan, et al. A P300-based brain–computer interface for people with amyotrophic lateral sclerosis. *Clinical Neurophysiology*, 119(8):1909–1916, 2008b.

F. Nijboer, N. Birbaumer, and A. Kubler. The influence of psychological state and motivation on brain-computer interface performance in patients with amyotrophic lateral sclerosis – a longitudinal study. *Frontiers in Neuroscience*, 4, 2010.

A. Nijholt. BCI for Games: A 'State of the Art' Survey. In *Entertainment Computing*, ICEC 2008, pages 225–228, 2008.

A. Nijholt, B. Reuderink, and D. Oude Bos. Turning Shortcomings into Challenges: Brain-Computer Interfaces for Games. *INTETAIN 2009, LNICST 9*, pages 153–168, 2009.

D.A. Norman. *The Design of Everyday Things*. MIT Press, 1998.

Neuper C. Guger C. Pfurtscheller G. Obermaier, B. Information transfer rate in a five-classes brain-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 9(3):283–288, 2001.

Fitzpatrick G. Buchannan-Dick M. McKeown J. O'Connor, C. Exploratory prototypes for video: interpreting pd for a complexly disabled participant. In *4th Nordic conference on Human-computer interaction: changing roles*, NORDICHI'06, pages 232–241, 2006.

The Oxford English Dictionary Online. URL `http://oxforddictionaries.com`.

T.L.B. Pape, J. Kim, and B. Weiner. The shaping of individual meanings assigned to assistive technology: a review of personal factors. *Disability and Rehabilitation*, 24(1): 5–20, 2002.

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

S. Perdikis, H. Bayati, R. Leeb, and J. Millán. Evidence accumulation in asynchronous BCI. *International Journal of Bioelectromagnetism*, 13(3):131–132, 2011.

Birbaumer N. Perelmouter, J. A binary spelling interface with random errors. *IEEE Transactions on Rehabilitation Engineering*, 8(2), 2000.

G. Pfurtscheller and C. Neuper. Dynamics of sensorimotor oscillations in a motor task. In B. Graimann, G. Pfurtscheller, and B.Z. Allison, editors, *Brain-Computer Interfaces*, The Frontiers Collection, pages 47–64. Springer, 2010.

G. Pfurtscheller and T. Solis-Escalante. Could the beta rebound in the EEG be suitable to realize a "brain switch"? *Clinical Neurophysiology*, 120(1):24–29, 2009.

G. Pfurtscheller, C. Neuper, and N. Birbaumer. Human brain-computer interface. *Motor cortex in voluntary movements*, pages 367–401, 2005.

G. Pfurtscheller, B.Z. Allison, C. Brunner, G. Bauernfeind, T. Solis-Escalante, R. Scherer, T.O. Zander, G. Mueller-Putz, C. Neuper, and N. Birbaumer. The hybrid bci. *Frontiers in Neuroscience*, 4(30), 2010.

Poel-M. Nijholt A. Plass-Oude Bos, D. A study in user-centered design and evaluation of mental tasks for bci. In *Proceedings of the 17th International Multimedia Modeling Conference*, MMM 2011, 2011.

A. Polaine. The flow principle in interactivity. *Proceedings of the Second Australasian*

Conference on Interactive Entertainment, pages 151–158, 2005.

D. Poulson, M. Ashby, and S. Richardson, editors. *USERfit: A practical handbook on user-centred design for rehabilitation and assistive technology.* HUSAT Research Institute for the European Commission, 1996.

G. Pullin. *Design Meets Disability.* Mit Press, 2009.

M. Quek, D. Boland, J. Williamson, R. Murray-Smith, M. Tavella, S. Perdikis, M. Schreuder, and M. Tangermann. Simulating the feel of brain-computer interfaces for design, development and social interaction. In *SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 25–28. ACM, 2011.

M. Quek, J. Höhne, R. Murray-Smith, and M. Tangermann. Designing future bcis: Beyond the bit rate. In A.Z. Brendan, S. Dunne, R. Leeb, J.R. Millan, and A. Nijholt, editors, *Towards Practical Brain-Computer Interfaces*, Biological and Medical Physics, Biomedical Engineering, pages 173–196. Springer, 2013.

M.G. Quiones. Listening in shuffle mode. *Song and Popular Culture*, 52:11–22, 2007.

A. Rauber and M. Frühwirth. Automatically analyzing and organizing music archives. In *Proceedings of the 5th European Conference on Research and Advanced Technology for Digital Libraries*, pages 402–414, 2001.

N. Ravaja. Contributions of psychophysiology to media research: Review and recommendations. *Media Psychology*, 6(2):193–235, 2004.

N. Ravaja and J. Kivikangas. Psychophysiological digital game playing: The relationship of self-reported emotions with phasic physiological responses. In *Measuring Behavior*, 2008.

A. Riccio, D. Mattia, L. Simione, M. Olivetti, and F. Cincotti. Eye-gaze independent eeg-based brain-computer interfaces for communication. *Journal of Neural Engineering*, 9(4), 2012.

B. Roark, J.d. Villiers, C. Gibbons, and M. Fried-Oken. Scanning methods and language modeling for binary switch typing. In *Proceedings of the NAACL HLT 2010 Workshop on Speech and Language Processing for Assistive Technologies*, SLPAT '10, pages 28–36, 2010.

R. Ron-Angevin and A. Díaz-Estrella. Brain-computer interface: Changes in performance using virtual reality techniques. *Neuroscience Letters*, 449(2):123–127, 2009.

S. Saeedi, R. Chavarriaga, J.d.R. Millán, and M.C. Gastpar. Prediction of fast and slow delivery of mental commands in bci. In *TOBI Workshop IV, Sion, Switzerland*, 2013.

E.B.N. Sanders. *From User-Centered to Participatory Design Approaches*, volume Design and the Social Sciences. Taylor & Francis Books Limited, 2002.

F.E. Sandes. Evaluating mobile text entry strategies with finite state automata. In *Proceedings of the 7th international conference on Human computer interaction with mobile devices services*, MobileHCI '05, pages 115–121, 2005.

R.G. Sargent. Verification and validation of simulation models. In *Winter Simulation Conference (WSC)*, pages 166 –183, 2010.

I. Saunders and S. Vijayakumar. The role of feed-forward and feedback processes for closed-loop prosthesis control. *Journal of NeuroEngineering and Rehabilitation*, 8(1):60, 2011.

J. Sauro and J.R. Lewis. Average task times in usability tests: What to report? In *SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 2347–2350, 2010.

G. Schalk, J.R. Wolpaw, D.J. McFarland, and G. Pfurtscheller. EEG-based communication: presence of an error potential. *Clinical Neurophysiology*, 111(12):2138–2144, 2000.

J.P.M. Schalkwijk. A class of simple and optimal strategies for block coding on the binary symmetric channel with noiseless feedback. *IEEE Transactions on Information Theory*, IT-17(3), 1971.

M.J. Scherer, C. Sax, A. Vanbiervliet, L.A. Cushman, and J.V. Scherer. Predictors of assistive technology use: The importance of personal and psychosocial factors. *Disability and Rehabilitation*, 27(21):1312–1331, 2005.

R. Scherer, A. Schloegl, F. Lee, H. Bischof, J. Jana, and G. Pfurtrscheller. The self-paced graz brain-computer interface: Methods and applications. *Computational Intelligence and Neuroscience*, 2007.

R. Scherer, F. Lee, A. Schlögl, R. Leeb, H. Bischof, and G. Pfurtscheller. Towards self-paced brain-computer communication: navigation through virtual worlds. *IEEE Transactions on Biomedical Engineering*, 55(2 Part 1):675–682, 2008.

A Schmidt. *Interactive context-aware systems interacting with ambient intelligence*, volume Ambient Intelligence. IOS, 2005.

Höhne J. Treder-M. Blankertz B. Tangermann M. Schreuder, M. Performance optimization of erp-based bcis using dynamic stopping. In *IEEE Engineering in Medicine and Biology Society*, 2011.

M. Schreuder, M. Tangermann, and B. Blankertz. Initial results of a high-speed spatial auditory bci. *International Journal of Bioelectromagnetism*, 11(2):105–109, 2009.

L.W. Schruben. Establishing the credibility of simulations. *Simulation*, 34(3):101–105, 1980.

R.L. Schwartz and T. Christiansen. *Learning Perl*. O'Reilly & Associates, 1997.

P. Shenoy, M. Krauledat, B. Blankertz, R.P.N. Rao, and K.R. Müller. Towards adaptive classification for BCI. *Journal of Neural Engineering*, 3(1):R13–R23, 2006.

T.B. Sheridan and W.R. Ferrell. *Man-machine systems*. MIT Press, 1974.

M. Silfverberg, I.S. MacKenzie, and P. Korhonen. Predicting text entry speed on mobile phones. In *SIGCHI Conference on Human Factors in Computing Systems*, pages 9–16, 2000.

C. Smith and H.I. Christensen. A minimum jerk predictor for teleoperation with variable time delay. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5621–5627, 2009.

R. Smith. Simulation, 1998. URL `http://www.modelbenders.com/encyclopedia/encyclopedia.html`.

S.W. Smith. *Digital Signal Processing: A Practical Guide for Engineers and Scientists*. Newnes, 2002.

C. Snyder. *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces*. Morgan Kaufmann, 2003.

R. Storn and E. Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997.

S. Strachan, R. Murray-Smith, and S. O'Modhrain. Bodyspace: inferring body pose for natural control of a music player. In *SIGCHI Conference on Human Factors in Computing Systems*, pages 2001–2006, 2007.

A. Sutcliffe, S. Fickas, M.M. Sohlberg, and L.A. Ehlhardt. Investigating the usability of assistive user interfaces. *Interacting with Computers*, 15:577–602, 2003.

P. Szekely. User interface prototyping: Tools and techniques. In *Software Engineering and Human-Computer Interaction*, pages 76–92. Springer, 1995.

J.d.R. Millán T. Carlson, G. Monnard. Vision-based shared control for a bci wheelchair. *International Journal of Bioelectromagnetism*, 13(1):20–21, 2011.

M. Tangermann, J. Höhne, M. Schreuder, M. Sagebaum, B. Blankertz, A. Ramsay, and R. Murray-Smith. Data driven neuroergonomic optimization of bci stimuli. In *5th Inter-*

national BCI Conference, Graz, 2011.

Müller K.R Aertsen A. Birbaumer N. Braun C. Brunner C. Leeb R. Mehring C. Miller-K.J. Müller-Putz G.R. Nolte G. Pfurtscheller G. Preiss H. Schalk G. Schlögl A. Vidaurre C. Waldert S. Blankertz B. Tangermann, M. Review of the bci competition iv. *Frontiers in Neuroprosthetics*, 6(55), 2012.

Schreuder M. Dähne S. Höhne J. Regler S. Ramsay A. Quek M. Williamson J. Murray-Smith R. Tangermann, M. Optimized stimulation events for a visual erp bci. *International Journal of Bioelectromagnetism*, 13(3), 2011.

D.M. Taylor and A.B. Schwartz. Direct cortical control of 3D neuroprosthetic devices, August 2004. US Patent App. 10/495,207.

L.G. Terveen. An overview of human-computer collaboration. *Knowledge Based Systems*, 8(2):67–81, 1995.

H. Thimbleby. Usability analysis with markov models. *ACM Transactions on Computer-Human Interaction, TOCHI*, 8(2):99–132, 2001.

H. Thimbleby. *Press on: principles of interaction programming.* MIT Press, 2007.

H. Thimbleby. Understanding User Centred Design (UCD) for People with Special Needs. *Computers Helping People with Special Needs*, pages 1–17, 2008.

L. Tonin, T. Carlson, R. Leeb, and J.R. Millán. Brain-controlled telepresence robot by motor-disabled people. In *23rd Annual International Conference of the IEEE EMBS*, pages 4227–4230, 2011.

G. Townsend, B. Graimann, and G. Pfurtscheller. Continuous EEG classification during motor imagery-simulation of an asynchronous BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 12(2):258–265, 2004.

N. Tractinsky and M. Hassenzahl. Arguing for aesthetics in human-computer interaction. *i-com*, 4(3):66–68, 2005.

N. Tractinsky, A. Katz, and D. Ikar. What is beautiful is usable. *Interacting with computers*, 13(2):127–145, 2000.

D. Travis. Measuring satisfaction: Beyond the usability questionnaire. URL `http://www.userfocus.co.uk/articles/satisfaction.html`.

Birbaumer N. Tregoubov, M. On the building of binary spelling interfaces for augmentative communication. *IEEE Transactions on Biomedical Engineering*, 2005.

A.M. Turing. Computing machinery and intelligence. *Mind*, 59:433–460, 1950.

Kahneman D. Tversky, A. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974.

A. Tychsen. Crafting user experience via game metrics analysis. In *Research Goals and Strategies for Studying User Experience and Emotion" at the 5th Nordic Conference on Human-computer Interaction: Building Bridges*, NORDICHI'08, pages 20–22, 2008.

G. Tzanetakis, M.S. Benning, S.R. Ness, D. Minifie, and N. Livingston. Assistive music browsing using self-organizing maps. In *2nd International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '09, pages 3:1–3:7. ACM, 2009.

Plass-Oude Bos D. Reuderink B. Poel M. Nijholt A. van de Laar, B. How much control is enough? optimizing fun with unreliable input, 2011.

M. van Gerven, J. Farquhar, R. Schaefer, R. Vlek, J. Geuze, A. Nijholt, N. Ramsey, P. Haselager, L. Vuurpijl, S. Gielen, and P. Desain. The brain-computer interface cycle. *Journal of Neural Engineering*, 6, 2009.

G. Vanacker, J.R. Millán, E. Lew, P.W. Ferrez, F.G. Moles, J. Philips, H. Van Brussel, and M. Nuttin. Context-based filtering for assisted brain-actuated wheelchair driving. *Computational Intelligence and Neuroscience*, 2007.

V. Venema, F. Ament, and C. Simmer. A stochastic iterative amplitude adjusted fourier transform algorithm with improved accuracy. *Nonlinear Processes in Geophysics*, 13(3): 321–328, 2006.

B. Venthur and B. Blankertz. A Platform-Independent Open-Source Feedback Framework for BCI Systems. In *Proceedings of the 4th International Brain-Computer Interface Workshop and Training Course*, 2008.

B. Venthur, S. Scholler, J. Williamson, S. Dähne, M.S. Treder, M.T. Kramarek, K.R. Müller, and B. Blankertz. Pyff–A pythonic framework for feedback applications and stimulus presentation in neuroscience. *Frontiers in Neuroscience*, 4, 2010.

L. Vertelney. Using video to prototype user interfaces. *ACM SIGCHI Bulletin*, 21(2):61, 1989.

C. Vidaurre and B. Blankertz. Towards a cure for bci illiteracy. *Brain Topography*, 23(2): 194–198, 2010.

R. A. Virzi, J. L. Sokolov, and D. Karis. Usability problem identification using both low- and high-fidelity prototypes. In *SIGCHI conference on Human Factors in Computing*

Systems, CHI '96, pages 236–243, 1996.

Y. Visell. Tactile sensory substitution: Models for enaction in HCI. *Interacting with Computers*, 21(1-2):38–53, 2009.

C. Wang, C. Guan, and H. Zhang. P300 brain–computer interface design for communication and control applications. In *International Conference IEEE Engineering in Medicine and Biology Society*, pages 5400–5403, 2005.

M.P. Ware, P.J. McCullagh, A. McRoberts, G. Lightbody, C. Nugent, G. McAllister, M.D. Mulvenna, E. Thomson, and S. Martin. Contrasting levels of accuracy in command interaction sequences for a domestic brain-computer interface using ssvep. In *5th Cairo International Biomedical Engineering Conference*, IBEC '10, pages 150–153, dec. 2010.

Wikipedia. User centered design. http://en.wikipedia.org/wiki/user-centered$_d$*esign*.

J. Williamson. *Continuous Uncertain Interaction*. PhD thesis, 2006.

J. Williamson, R. Murray-Smith, and S. Hughes. Shoogle: Excitatory multimodal interaction on mobile devices. In *Proceedings of the SIGCHI Conference on Human factors in Computing Systems*, CHI '07, pages 121–124, 2007.

J. Williamson, R. Murray-Smith, B. Blankertz, M. Krauledat, and K. Müller. Designing for uncertain, asymmetric control: Interaction design for brain-computer interfaces. *International Journal of Human-Computer Studies*, 67(10):827–841, 2009.

J.R. Wolpaw and D.J. McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences*, 101(51):17849, 2004.

J.R. Wolpaw, N. Birbaumer, D.J. McFarland, G. Pfurtscheller, and T.M. Vaughan. Brain–computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767–791, 2002.

G.N. Yannakakis. How to model and augment player satisfaction: A review. In *Proceedings of the 1st Workshop on Child, Computer and Interaction. ICMI '08*, 2008.

G.N. Yannakakis and J. Hallam. Towards optimizing entertainment in computer games. *Applied Artificial Intelligence*, 21(10):933–972, 2007.

G.N. Yannakakis and J. Hallam. Entertainment modeling through physiology in physical play. *International Journal of Human-Computer Studies*, 66(10):741–755, 2008.

N. Yeung, R. Bogacz, C.B. Holroyd, and J.D. Cohen. Detection of synchronized oscillations

in the electroencephalogram: an evaluation of methods. *Psychophysiology*, 41(6):822–832, 2004.

J. Yue, Z. Zongtan, J. Jiang, Y. Liu, and D. Hu. Balancing a simulated inverted pendulum through motor imagery: An eeg-based real-time control paradigm. *Neuroscience Letters*, 524(2):95–100, 2012.

T.O. Zander and C. Kothe. Towards passive brain–computer interfaces: applying brain–computer interface technology to human–machine systems in general. *Journal of Neural Engineering*, 8, 2011.

D. Zhang, Y. Wang, A. Maye, A.K. Engel, X. Gao, B. Hong, and S. Gao. A Brain-Computer Interface Based on Multi-Modal Attention. In *Proceedings of the 3rd International IEEE/EMBS Conference on Neural Engineering*, pages 414–417, 2007.

H. Zhang, C. Guan, and C. Wang. Asynchronous P300-based brain-computer interfaces: a computational approach with statistical models. *IEEE Transactions on Biomedical Engineering*, 55(6):1754–1763, 2008.