A HYBRID BIOLOGICAL/IN SILICO NEURAL NETWORK BASED BRAIN-MACHINE INTERFACE

by

Mehmet Kocatürk

BS, Electronics and Communication Engineering, Istanbul Technical University, 2004

MS, Biomedical Engineering, Boğaziçi University, 2007

Submitted to the Institute of Biomedical Engineering

in partial fulfillment of the requirements

for the degree of

Doctor

of

Philosophy

Boğaziçi University

2015

A HYBRID BIOLOGICAL/IN SILICO NEURAL NETWORK BASED BRAIN-MACHINE INTERFACE

APPROVED BY:

Prof. Dr. Albert Güveniş .....
(Thesis Advisor)

Prof. Dr. Halil Özcan Gülçür .....
(Thesis Co-advisor)

Prof. Dr. Reşit Canbeyli .....

Assoc. Prof. Dr. Burak Güçlü .....

Assist. Prof. Dr. Bora Garipcan .....

Prof. Dr. Gürkan Öztürk .....

DATE OF APPROVAL: 20 May 2015

Academic Ethics and Integrity Statement

(To be completed upon submission of Master and PhD Theses for examination)

I am familiar with the YÖK (Higher Council of Education) Academic Ethics and Integrity Policy and I understand the potential consequences should my thesis be found to contain plagiarized content or violate this policy in any other way.

Name: ..... Date: .....

Signature: .....

ACKNOWLEDGMENTS

I would like to express my sincere thanks to Prof. Albert Güveniş and Prof. Halil Özcan Gülçür for their guidance and generous reliance on me to perform research in the field of neural engineering. Prof. Güveniş has always been available to help and guide me to complete this work. It has been an honour for me to discuss every detail of our findings with Prof. Gülçür. I highly appreciate their guidance and neverending patience in our discussions.

I am tremendously indebted to Prof. Canbeyli for his guidance, thorough and always constructive criticisms, and outstanding patience to improve the quality of the present work. It would never be possible for me to see the results of the present work without being a member of his team consisting of highly talented and innovative researchers. I would like to express my special thanks to my colleagues Onur İyilikçi, Aysu Mutlutürk, Emre Laçin, Elçin Tunçkol and Başak Alpat for their unique friendship and support in the present study.

I am deeply grateful to Assoc. Prof. Burak Güçlü for his invaluable contributions to this work. Dr. Güçlü has significantly improved the comprehension of this work not only with his constructive comments but also by providing equipments for the experimental setup.

I would also like to express my sincere thanks to Prof. Gürkan Öztürk and Assist. Prof. Bora Garipcan for their participation in the thesis committee and helpful suggestions for further improvements in the neuroprosthetic system described in the present dissertation.

This work has been supported by Boğaziçi University BAP Grants #10XD3 and Bogazici University Life Sciences and Technologies Research Center #09K120520.

ABSTRACT

A HYBRID BIOLOGICAL/IN SILICO NEURAL NETWORK BASED BRAIN-MACHINE INTERFACE

Brain-machine interfaces (BMIs) aim to improve the lives of individuals with neurological disease or injury, by opening new information transfer channels between brain tissue and prosthetic actuators. In a majority of the BMI work, the data acquired from the motor cortex neurons are decoded into user's intended prosthetic actions by some "optimized" input-output mathematical model. Although this approach is quite sound, the information processing principles used are fundamentally different from those of natural neural circuits. In this thesis, we propose a novel, neurally-inspired design approach; the BMI controller consists of spiking model neurons and receives simulated synaptic inputs from extracellularly recorded neurons. The controller therefore forms a hybrid biological/*in silico* neural network with the neuronal circuits of the user's brain. In order to fulfill the challenging real-time requirements of the present design approach, we first developed the Bioinspired Model Development Environment (BMDE). The BMDE, implemented on a hard real-time system, significantly facilitates BMI model development processes with powerful online data visualization tools while satisfying the strict timing constraints of the proposed design approach. Using the BMDE, we realized a novel, adaptive BMI controller which consists of *in silico* striatal medium spiny neurons, each receiving simulated synaptic inputs from extracellularly recorded motor cortex neurons. By implementing a reward-modulated spike timing-dependent plasticity rule and a winner-takes-all mechanism, the BMI controller, based on real-time closed-loop simulations, achieves perfect target reach accuracy for a two target reaching task in one dimensional space. Using this design approach and the BMDE, new generation BMI controllers that better mimic brain circuits can be developed. Moreover, by investigating the interactions between biological and *in silico* neural networks during neuroprosthetic control tryouts new neuroscientific insights concerning motor control and learning can be obtained.

Keywords: Neuroprosthetics, Motor Cortex, Synaptic Plasticity Model.

ÖZET

HİBRİT BİYOLOJİK/İN SİLİCO SİNİR AĞI TEMELLİ BİR BEYİN-MAKİNE ARAYÜZÜ

Beyin-makine arayüzleri (BMA) beyin dokusu ile protetik hareket düzenekleri arasında yeni bilgi iletim kanalları açarak nörolojik hastalık veya yaralanmaya maruz kalmış bireylerin yaşamlarını iyileştirmeyi hedefler. BMA çalışmalarının çoğunda, motor korteksten alınan veri kullanıcının gerçekleştirmeyi hedeflediği protetik eylemlere bazı "eniyilenmiş" girdi-çıktı matematiksel modeller kullanılarak dönüştürülür. Bu yaklaşım çok güvenilir olsa da, kullanılan bilgi işleme prensipleri doğal sinirsel devrelerinkinden temelde farklıdır. Bu tezde, yeni, sinir devrelerinden esinlenen bir tasarım yaklaşımı önermekteyiz; BMA kontrolcüsü model nöronlardan oluşmakta ve elektrofizyolojik yöntemlerle aktivitesi kaydedilen nöronlardan benzetilen sinaptik girdiler almaktadır. Dolayısıyla, kontrolcü kullanıcının beyin devreleri ile bir hibrit biyolojik/ *in silico* sinir ağı oluşturmaktadır. Bu tasarım yaklaşımının zorlu gerçek-zaman gereksinimlerini karşılamak için, öncelikli olarak Biyoesinlenmiş Model Geliştirme Ortamı (BMDE)'nı geliştirdik. Gerçek-zamanlı bir sistem üzerinde meydana getirilmiş olan BMDE önerilen tasarım yaklaşımının sıkı zamanlama şartlarını yerine getirmenin yanı sıra güçlü çevrimiçi veri görselleştirme araçlarıyla BMA model geliştirme işlemlerini önemli ölçüde kolaylaştırmaktadır. BMDE'yi kullanarak her biri gerçek motor korteks nöronlarından benzetilen sinaptik girdiler alan *in silico* striatum ortaboy dikenli nöronlarından oluşan bir BMA kontrolcüsü geliştirdik. BMA kontrolcüsü, gerçek-zamanlı benzetimler temelinde, bir ödülle değişen vuru zamanlamasına bağlı sinaptik plasitisite kuralını ve bir "kazanan hepsini alır" mekanizmasını uygulayarak tek boyutlu uzayda iki farklı hedefe kusursuz şekilde ulaşma başarımını göstermektedir. Bu tasarım yaklaşımını ve BMDE'yi kullanarak beyni daha iyi taklit edebilen BMA kontrolcüleri geliştirilebilir; nöroprotetik kontrol denemeleri sırasında biyolojik ve *in silico* sinir ağları arasındaki etkileşimler gözlemlenerek motor kontrol ve öğrenmeye ilişkin yeni sinirbilimsel bilgiler elde edilebilir.

Anahtar Sözcükler: Nöroprotez, Motor Korteks, Sinaptik Plastisite Modeli.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a_{ij}	Description of a_{ij}
α	Description of α
v	Membrane potential
u	Membrane recovery variable
C	Membrane capacitance
a	Constant describing time scale of u
b	Constant describing the sensitivity of u
k	Scaling constant for Izhikevich neuron model
η	Total excitatory synaptic conductances
γ	Total inhibitory synaptic conductances
E_η	Excitatory synaptic reversal potentials
E_γ	Inhibitory synaptic reversal potentials
τ_η	Time constant for the decay of excitatory synaptic conductance
τ_γ	Time constant for the decay of excitatory synaptic conductance
v_{peak}	Peak membrane voltage
c	Membrane potential reset value after a spike generation
d	Membrane recovery variable reset value after a spike generation
$w_{ij(t)}$	Weight of the synapse between the i-th and j-th neuron
μ	Learning rate
$e_{ij(t)}$	Eligibility trace of the synapse between the i-th and j-th neuron
$r(t)$	Global reward signal
$S(t)$	Sensory error
$\bar{R}_k$	Positive reward estimate for the k-th target
n_k	Trial number for the k-th target

R_T	Binary reward variable
m	Width of the averaging window
W	Sum of all weights of excitatory synapses to a medium spiny neuron
w_{max}	Maximum value for excitatory synapses
α	Scaling constant for w_{max}
N_j	Total number of excitatory synapses to the j-th medium spiny neuron
t	Time variable
x	32-bit number generated by the pseudo-random number generator
s	Iteration counter for the pseudo-random number generator

LIST OF ABBREVIATIONS

ADC	Analog-to-digital conversion
API	Application programming interface
B-BMI	Bioinspired brain-machine interface
BMDE	Bioinspired Model Development Environment
BMI	Brain-machine interface
COMEDI	Linux Control and Measurement Device Interface
CPU	Central processing Unit
DAQ	Data acquisition
DIO	Digital input output
GABA	Gamma aminobutyric acid
GL	Guide LED
GPe	Globus pallidus externa
GPi	Globus pallidus interna
GUI	Graphical user interface
ICD	In-circuit Debugger
LED	Light Emitting Diode
LTD	Long term depression
LTP	Long term potentiation
M1	Primary motor cortex
MCU	Microcontroller unit
MSN	Medium spiny neuron
MWA	Microwire array
PC	Personal computer
PCA	Principle component analysis
PCB	Printed circuit board
PVA	Population vector algorithm
RT	Real-time
RTAI	Real-time Application Interface for Linux

SNc	Subtantia nigra pars compacta
SNN	Spiking neural network
SNr	Subtantia nigra pars reticulata
STDP	Spike-timing-dependent plasticity
STN	Subthalamic nucleus
TL	Target LED
TP	Task period
TTL	Transistor-transistor logic
WTA	Winner-take-all

1. INTRODUCTION

1.1 Neuroprosthetics Overview

Research for neuroprosthetics, or brain-machine interfaces (BMIs), have recently allowed tetraplegic individuals to control a robotic arm [1, 2] or a computer cursor [3] directly with their motor cortical activity. These clinical trials have been important proof-of-concept demonstrations for restoring the lost motor functions of the central nervous system using brain implantable devices. Even though such seminal devices have enabled the patients to manipulate their environment through prosthetic movements, further research is still ongoing to deliver seamless, robust and high performance solutions.

First intentional neuroprosthetic control was achieved in 1960s through neuronal operant conditioning, in which the subject (a rat) was rewarded when the activity of an extracellularly recorded neuron increased without any overt movement [4, 5]. In 1980s, Georgopoulos et al. showed in monkeys that there is an almost linear relationship between firing rates of some motor cortical neurons and arm movement directions in a center-out reaching task [6, 7]. The same group also demonstrated that a neuronal population vector, yielded by summation of the contribution of each neuron to the movement direction, can be used to accurately predict the hand trajectory from the neural recordings [8]. This decoding method is named as population vector algorithm (PVA). In 1999, Chapin et al. showed in rats that recordings from the neuronal populations can be utilized for closed-loop control of a robotic arm in one-dimensional space without any explicit limb movement [9]. In this study, principal components analysis method was used to reduce the dimension of the neural data and the first principal component was fed into a recurrent artificial neural network to control the one-degree-of-freedom robotic actuator. In 2002, Taylor et al. showed on non-human primates that a 3D neuroprosthetic device could be manipulated in real-time using the recordings from the motor cortex. They also showed that the target reach success of

the BMI user (the monkey) was higher in the closed-loop control mode than that in the open-loop mode while the same decoding algorithm (PVA) was utilized in both cases [10]. In closed-loop mode a visual feedback was provided to the user online and in the open-loop mode the trajectories were created offline from cortical signals recorded during hand-controlled cursor movements. In 2003, Carmena et al. utilized a Wiener filter to transform neural activity patterns into prosthetic actions and reported that the directional tuning properties of the neurons in the motor-related cortical areas were modified in the brain control mode as prosthetic control sessions proceeded [11]. After these significant contributions to the neuroprosthetic research, numerous works were realized to further boost the decoding performance using different input-output models [12, 13, 14, 15]. Since the structure of the motor cortical areas and activity patterns of cortical neurons continuously change through neuroplasticity [16, 17, 18, 19, 20], in some of these works the goal was to develop coadaptive decoders [21, 22, 23, 24, 25, 26] capable of modifying their model parameters to respond to the dynamics of tuning functions of motor cortical neurons.

In spite of these advancements, neuroprosthetic systems still lack self-adaptation. In other words, decoder calibration by a caregiver or an external feedback signal is still required. In addition, conventional decoders still cannot identify when to keep the prosthetic arm stationary (idle state) and when to manipulate it for reaching (active state) [27]; the reaching trials are manually initiated by the experimenter or the prosthetic user through physical movements. However, in a real clinical setting, the paralyzed individual will need to be able to manipulate the prosthesis only when he/she intends to use it. Consequently, according to the current state-of-the-art, precision, speed and robustness of prosthetic movements are still far from those of natural motor movements [28, 29].

1.2 Neurally-Inspired BMI Design - The Objective

As mentioned in the previous section and shown in Figure 1.1, conventional neuroprosthetic systems have been designed from the perspective of input-output mathe-

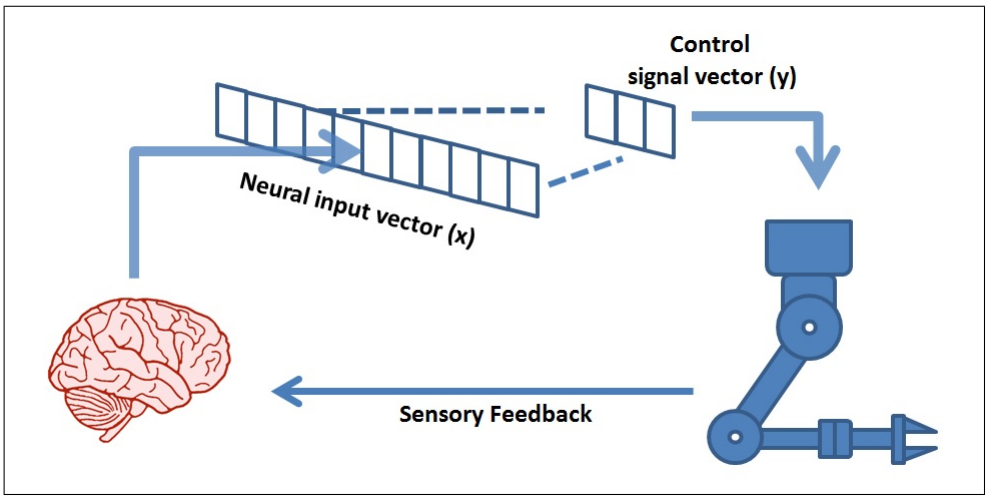

Figure 1.1 Conventional closed-loop BMI control architecture. An input-output model projects high-dimensional neural vector into low-dimensional control signal vector.

matical modeling; the main design motivation has generally been to find a mathematical model which optimally maps motor cortical activity to user's intended prosthetic actions. In addition, a spike binning process (Figure 1.2) is performed in order to provide cortical firing rate inputs to the utilized model or decoder. However this process leads to loss of information encoded by spike timing. In this context, information processing principles of these systems, based on an *input-output model* or *transform*, are fundamentally different from those of natural neural circuits.

In the present dissertation, it is argued that the BMI controllers could be formed from a more biologically plausible design perspective using spiking neural networks (SNNs). In such a control paradigm, the SNN consists of biologically plausible model neurons and receives simulated synaptic inputs from the extracellularly recorded cortical neurons. The controller therefore forms a hybrid biological/*in silico* neural network with the neuronal circuits of the user's brain. Its outputs are then used in manipulating a neuroprosthesis (Figure 1.3). Using this approach, the dynamics of the model neurons could be investigated during neuroprosthetic control experiments while they are directly interacting with real neurons. Moreover, novel neuroprosthetic control algorithms which are inspired by the neuronal circuits of the brain structures could be developed.

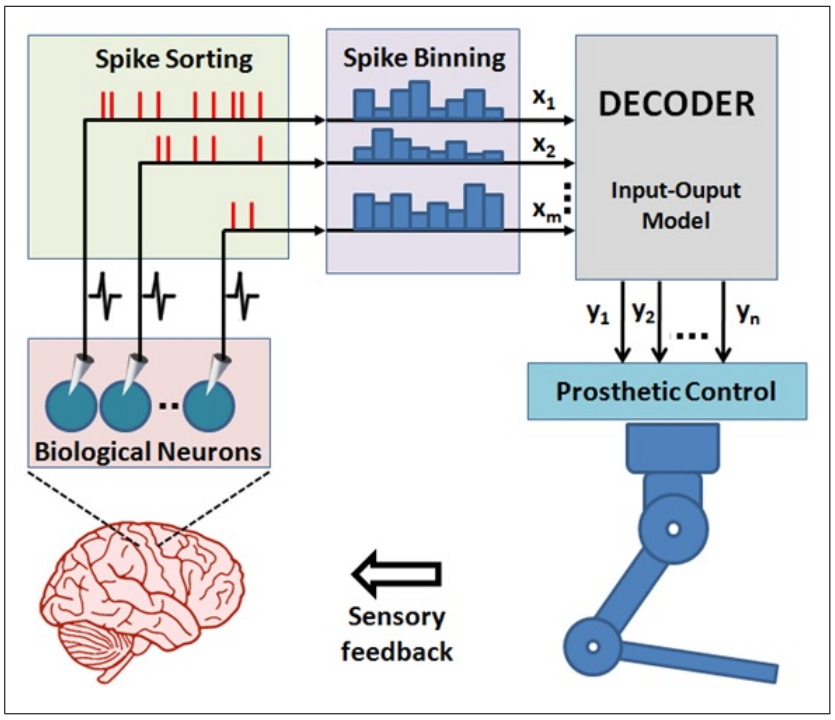

Figure 1.2 Spike-binning to provide firing rate inputs to the BMI decoder.

Despite offering promising methods for computational neuroscience and BMI research, SNN-based neuroprosthetic control paradigm requires powerful and purpose-specific platforms which are capable of 1) real-time SNN simulation, 2) providing biologically realistic synaptic interactions between real and *in silico* neurons and 3) manipulating a robotic actuator according to the spike outputs of *in silico* neurons in real-time. In this context, the main motivation of the work in the present study is to address these requirements of SNN-based neuroprosthetic design and show the proof-of-concept for the present approach by developing a brain machine interface (BMI) control algorithm which is inspired by the dynamic synaptic interactions between the motor cortex and striatum neurons. The present work does not attempt to compete with the control performance of existing input-output model-based neuroprosthetic systems. The focus of the present work is to demonstrate the feasibility of building hybrid biological-*in silico* neural networks for neuroprosthetic control and shift the neuroprosthetic design philosophy for development of brain-inspired control architectures. We believe that this novel, SNN-based design approach has the potential to bring several advantages in neuroprosthetic system control, adaptation and implementation. Firstly, the information encoded by spike timing could be used at the input layer of the SNN-

based BMI controller. Secondly, spike timing plays a critical role in neuroplasticity [30], which is essential in neuroprosthetic learning [31]. Therefore, the SNN-based BMI controllers updating their parameters by simulating mechanisms of spike timing-dependent plasticity might have superior adaptation performance than existing firing-rate based neuroprosthetic systems. Thirdly, implementation of the SNN-based BMI controllers into neuromorphic chips [32, 33] can enable delivery of fully implantable, ultra low power neuroprosthetic systems for paralyzed patients. In addition, the SNN-based design approach can also be beneficial in the field of neuroscience. The interactions of real neurons with model neurons could be investigated during neuroprosthetic control experiments and these investigations can provide new insights into the information processing principles in the motor cortex during neuroprosthetic control and learning.

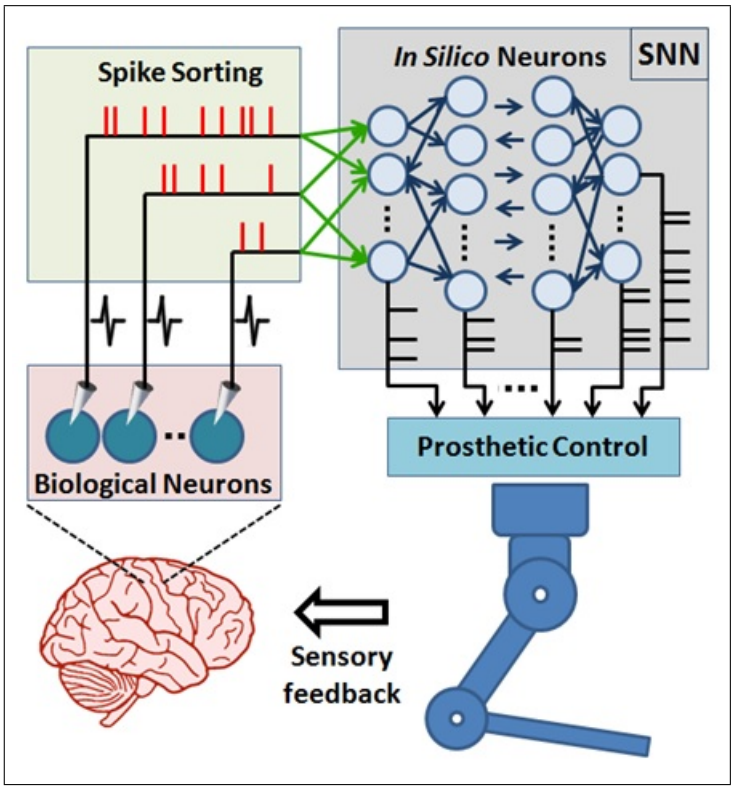

Figure 1.3 Principal components of BMI paradigm based on building a hybrid biological-*in silico* neural network. A spike sorting utility continuously acquires neural signals from the brain through extracellular recordings and extracts the timestamps of the spikes generated by real neurons. The spike events are then streamed to the biologically plausible model neurons as 'virtual' synaptic inputs and these inputs trigger further information processing in the spiking neural network (SNN). Finally, the prosthetic control module translates the spike event outputs of the SNN into prosthetic command signals for manipulation of the neuroprosthesis.

1.3 Overview of the Dissertation

Chapter 1 summarizes current state-of the art in motor neuroprosthetic design and the design perspective of neuroprosthetic systems. The novel, biologically inspired BMI controller design concept which is brought by the work in the present dissertation is also described.

Chapter 2 explains our established, low-cost techniques to perform chronic extracellular recordings from the rat motor cortex, which enabled us to see the requirements of development process of a biologically plausible BMI controller.

Chapter 3 presents the hardware and software components of the Bioinspired Model Development Environment (BMDE) which enables online data visualization while realizing the simulated synaptic interactions between the biological and *in silico* neurons and control of a robotic arm. This chapter also describes the in-house built robot and experimental environment controller, the control hardware.

Chapter 4 includes a brief overview of the motor and basal ganglia system and the role of striatum in the cortico-basal ganglia circuit. This chapter demonstrates the control architecture of the bioinspired BMI (the B-BMI), which is designed by utilizing the information processing principles and neuroanatomy of the corticostriatal circuit.

Chapter 5 demonstrates the proof-of-concept for the B-BMI by studying its convergence properties using real-time closed-loop simulations, in which the extracellular recordings from the motor cortex neurons are imitated by a hardware-based, external neural signal synthesizer. The closed-loop simulations also involved a behavioral paradigm implemented on the BMDE and interacting with the experimental environment through external triggers.

Chapter 6 shows the real-time performance profiles of the BMDE while running only the B-BMI and a stress test, which includes simultaneous running of the B-BMI and simulating 150 medium spiny neurons. These performance profiles provide an

idea about usability of the BMDE in development more sophisticated BMI controllers which based on a higher number of neurons or more complex neuroprosthetic control parameters.

Chapter 7 lists the novel contributions of the work in this dissertation and implications of them in helping paralyzed people. Future improvements in the BMDE and B-BMI are also presented in this chapter.

2. EXTRACELLULAR RECORDING FROM BEHAVING RAT

Prior to starting the studies for development of a BMI control algorithm, we concentrated on developing robust techniques for chronic extracellular recordings from the rat motor cortex. We built an experimental environment so that rats can perform motor tasks and we can verify the position of recording electrodes in the primary motor cortex area in the rat brain. Simultaneous to the efforts for experimental environment construction, we implemented the software and hardware required for neural recordings.

2.1 Techniques for Chronic Extracellular Recordings

Electrophysiology hardware is required in order to perform extracellular recordings from the brain. The hardware consists of microelectrode assemblies, signal filters and amplifiers. Figure 2.1 demonstrates the components utilized in our laboratory to enable 32 channel (spike-only) recordings. We utilize a 32 channel signal amplifier (PRA3) produced by Plexon Inc. (TX, USA) and a data acquisition device (PCIe-6259) provided by National Instruments (TX, USA). The software for digital signal filtering and recordings are developed by us in the laboratory and will be explained in Chapter 3 by introducing the Bioinspired Model Development Environment. The microwire arrays are also assembled in the laboratory using a technique we developed (Chapter 2.1.1).

2.1.1 Microwire Array Production

Microwire arrays (MWAs) are the intermediaries between the electronic and brain circuits. They are used for recording extracellular multi-unit spike activity from the neuronal populations. Teflon insulated platinum-iridium (%90-%10) wires with a

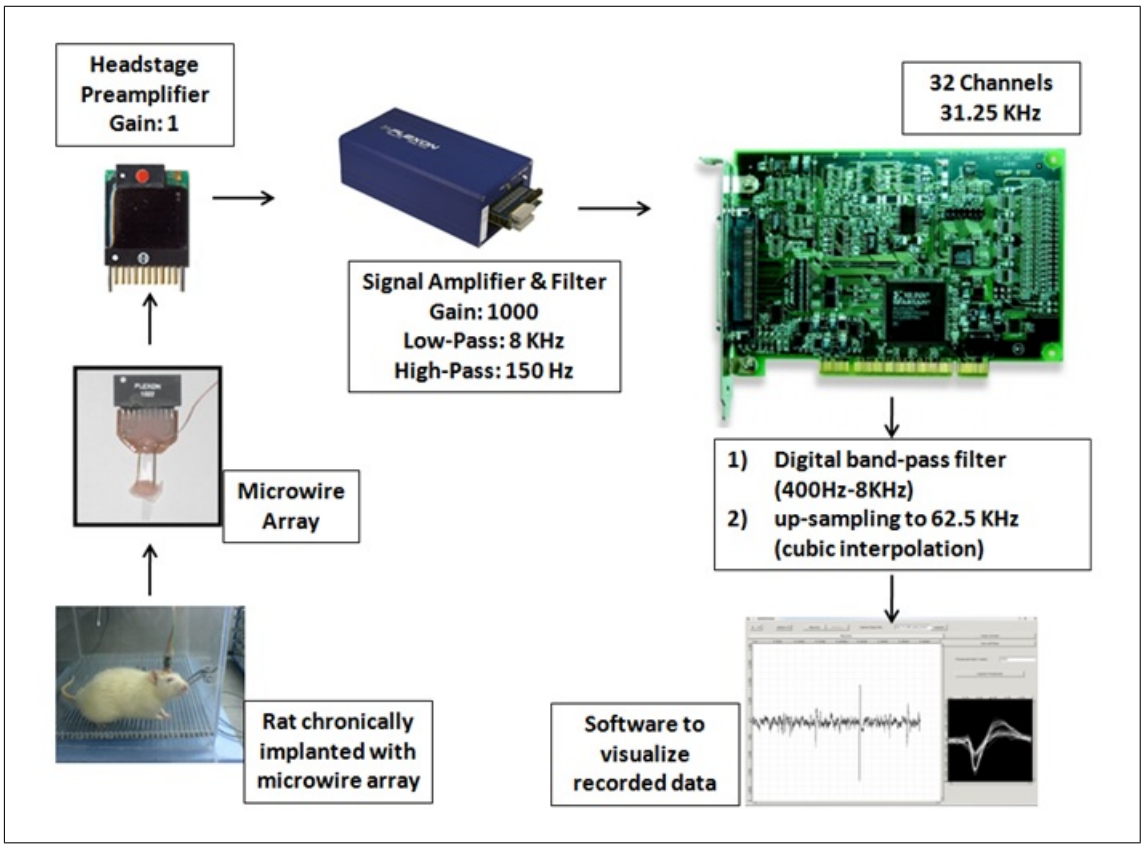

Figure 2.1 Generic components of an electrophysiological recording system.

diameter of 25.4 μm (A&M Systems Co, USA) are used in our experiments.

In the production process of the MWAs, spooled microwires are cut into 12 cm segments. Thereafter, two Kelly hemostats are clipped to each end of the curved wire segments. The position of the upper hemostat is fixed and the lower hemostat is allowed to be manually rotated until the wire becomes straight [34]. The straightened wires are then cut into 4 pieces to be aligned on the jig seen in Figure 2.2A. To build the jig, we use a microscope slide and two stainless steel templates produced by a printed circuit board (PCB) stencil manufacturer (BEK Lazer Elektronik, Istanbul). The stainless steel templates have slots with a spacing of 250 μm (Figure 2.2C). The width of each slot is identical and 50 μm. These templates are vertically mounted onto two opposite sides of the microscope slide so that the previously straightened wires can be aligned in parallel through their slots. A piece of paper on which parallel lines are printed with a distance of 250 μm is placed under the microscope slide to guide the

wire alignment process.

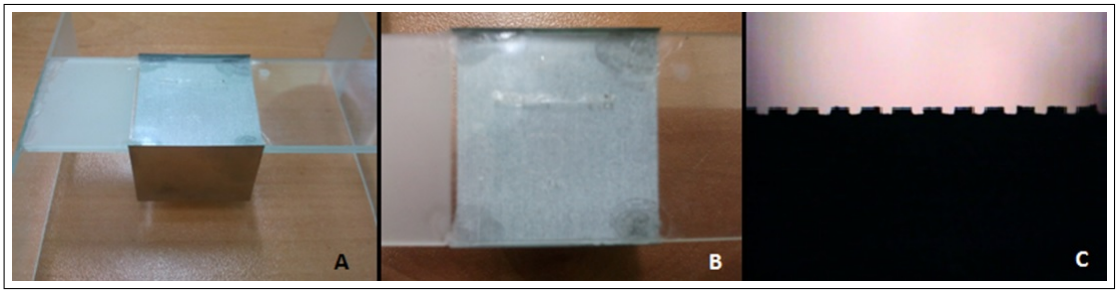

Figure 2.2 The rig for assembling microwire array. A) Side view of the rig. Two stainless steel templates are mounted on both sides of the microscope slide. B) Top view of the rig. C) The chamfers of the stainless steel template under microscope (40X zoom).

Eight insulated Pt/Ir wires for recording and one uninsulated tungsten wire to be used as a reference electrode are put into proper positions within the slots of the template of the jig, and a drop of dental acrylic is poured to the center of wires to stabilize the position of all wires. After dental acrylic sets, the wires are carefully removed from the template without disturbing their positions. Hereupon, the acrylic holding the wires is glued onto a PCB. A small connector (Omnetics 8o50m-10P, USA) is also soldered on this PCB to enable *in vivo* recordings (Figure 2.3). Finally the wires are soldered to the appropriate paths on the PCB and a ground wire is added to the MWA to be soldered to a ground screw placed into the skull of the rat during the implantation surgery.

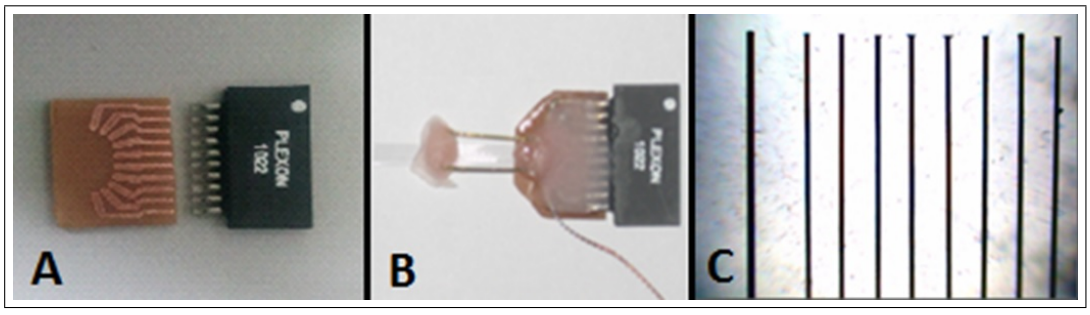

Figure 2.3 Eight-channel MWA. A) PCB & connector for the MWA. B) Assembled MWA, C) Channels of the MWA under microscope (40X zoom). The left most wire is the tungsten uninsulated tungsten wire used as a reference electrode.

To finalize the microwire preparation process, the tips of the wires of MWA are cut using sharp scissors to make the length of the wires equal. By using an impedance measurement device (World Precision Instruments - Omega-Tip Z, FL,

USA), the impedance of each wire is verified to be around 0.7-1.2 MΩ at 0.5 KHz in saline solution.

2.1.2 Microwire Array (MWA) Implantation

The caudal motor cortex forelimb area (Figure 2.4) has been shown to be predictive of limb movements and exhibits neuronal activity modulations for the control of a neuroprosthesis without physical movements [9, 31].Therefore, we have primarily targeted this area for neural recordings for developing BMI controllers. The in-house built, single row, 8 channel MWAs (Figure 2.3) are bilaterally implanted in this area using a stereotaxic apparatus. During the surgery, we adjust the depth of the MWAs to record from layer V of the motor cortex (Figure 2.5), which mainly includes large pyramidal neurons projecting to the spinal cord and the striatum [35].

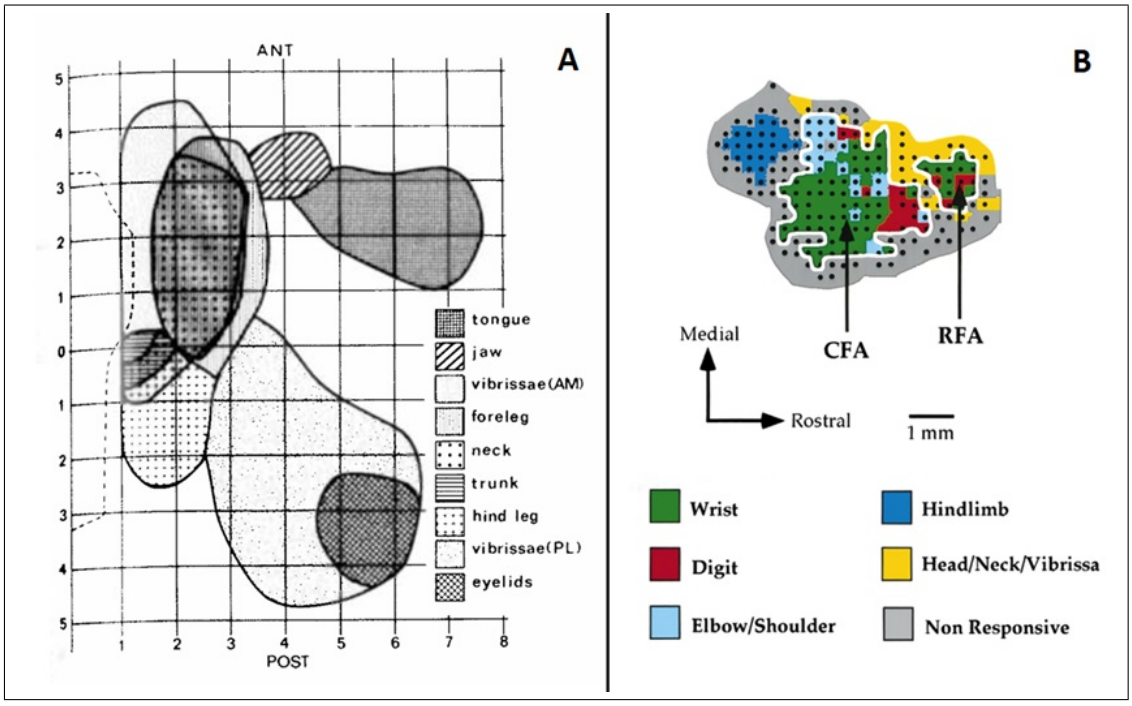

Figure 2.4 Motor areas in the rat brain based on electrical microstimulation. A) General motor map (from [36]). B) Detailed map of the forelimb area (from [16]).

The MWA is lowered into the brain as slowly as possible to minimize distress to the brain tissue until reaching a depth of approximately 1.2 mm [37]. After verification

of MWA position with electrophysiological monitoring, the surface of the brain is sealed using cyanoacrylate glue [31]. Four screws are mounted onto the skull. Dental cement is poured over the skull and around the screws to fix the position of the MWAs.

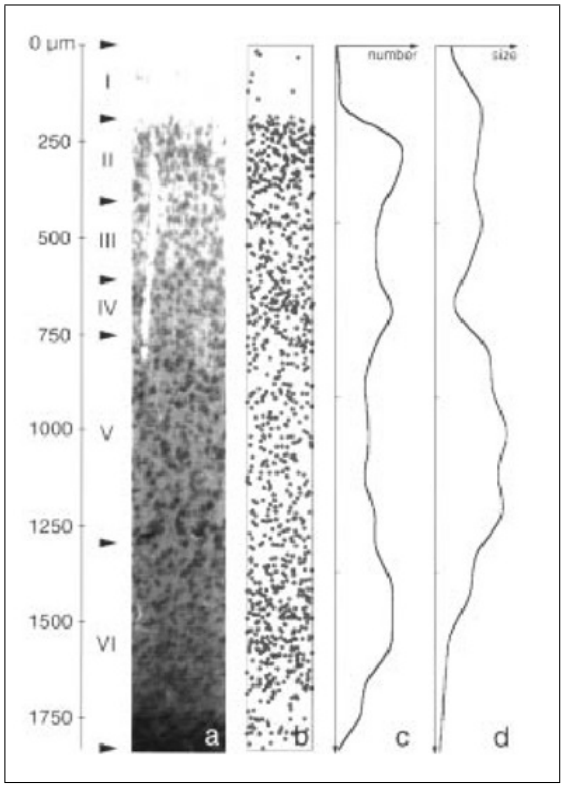

Figure 2.5 Layers of the primary motor cortex in the rat [37]. a) Micrograph, b) Reconstruction through image processing, c) Number of cells in layers, d) Size of the cells.

2.2 Building the Experimental Environment

Neuroprosthetic research requires not only powerful hardware and software platforms for development of control algorithms but also applicable behavioral paradigms which enable *in vivo* experimentation of the proposed control algorithms. In order to enable motor neuroprosthetic control research, we built an experimental environment, which mainly consists of mainly two components. The first component is an operant conditioning box while the second is a robotic arm which moves outside of the box, in the robotic workspace. The environment in this work was designed to be used in future *in vivo* BMI experiments. In this scenario, the rat is planned to be used as a paralyzed patient model.

In the experimental environment, the cage includes two levers, a water receptacle and an infrared beam (Figure 2.6). The robotic arm actuates a Guide LED in one dimensional space, to left and right. In the cage there are two LEDs and, in the robotic workspace, there are two Target LEDs (TLs). The box is made of Plexiglas so that the rat can see the robotic workspace.

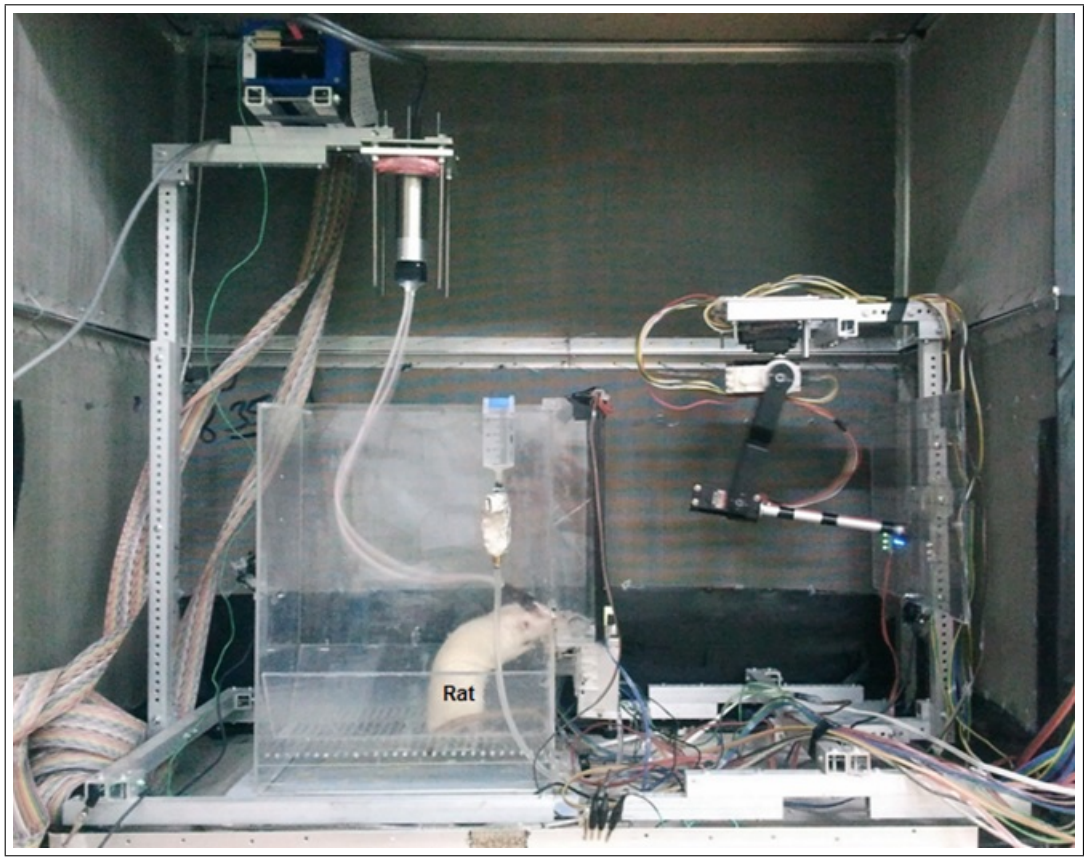

Figure 2.6 Experimental environment including a robotic arm.

2.3 Moving Attention of the Rat to the Robotic Workspace

The robotic arm was placed into the experimental environment as a neuroprosthesis. In order to investigate the neural activity of the rat in a two lever choice task and validate the usability of the experimental environment for neuroprosthetic control experiments, we implemented a behavioral paradigm as in [38].

In the paradigm, the rat initiates a trial by a nose poke through the IR beam

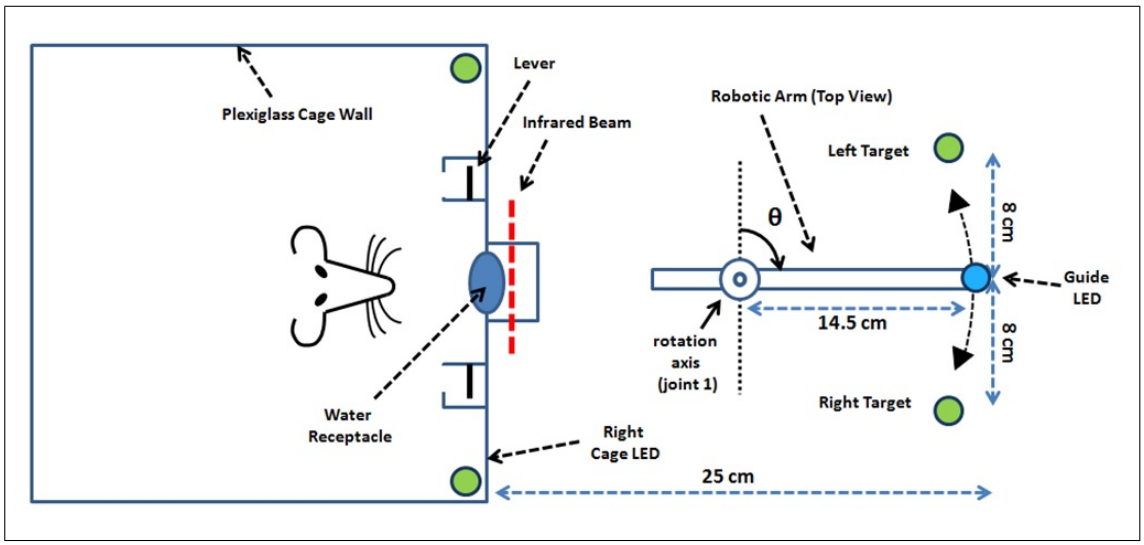

Figure 2.7 Representative top view of the experimental environment for a two lever choice task.

(Figure 2.8). Once the trial starts, randomly selected Target LED (TL), the Cage LED (CL) at the same side as TL and Guide LED (GL) are turned on. Without any latency after trial start, the GL, mounted on the tip of the robotic arm, moves toward the selected TL with a randomly selected speed and reaches it in 0.9-1.5 s. The rat is rewarded (0.03 ml water) only when it continuously presses the correct lever for 50 ms after GL reaches the TL. Here the correct lever is the lever which is in the same direction as the turned-on TL or CL. In this paradigm, the changing speed of GL encourages attention of the rat. The rat can minimize the task energy by synchronizing lever press with the arrival of the GL to the target. Whenever the rat makes a wrong lever choice, a brief tone is presented and the trial is terminated. When a trial ends, all the LEDs are turned off and GL moves back to the default position in the middle of the Target LEDs. As the rat increases its success rate in tasks, the CL is removed from experimental paradigm and the attention of rat is gradually moved to TL and GL, to robotic workspace. When the rat's accuracy exceeds an inclusion criterion of 80%, the shaping process is terminated and the rat is implanted with microelectrode arrays. Prior to the neural recordings, rats are given two weeks to recover from the surgery.

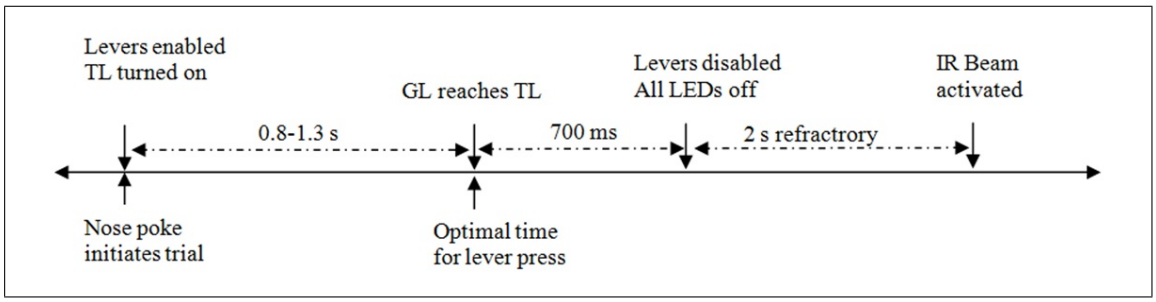

Figure 2.8 Rat training task.

2.4 Validating In Vivo Recording Techniques

Since the work in this dissertation was mainly focused on developing a brain-inspired neuroprosthetic control algorithm and a platform for designing such neuroprosthetic processors, we did not systematically investigated the interactions in the tissue-Pt/Ir electrode interface or report the well-known neural correlates of a lever press movement. There are a number of significant works that present these aspects of neuroprosthetic research [9, 39]. In this context, we mainly performed the recordings to validate our surgical, recording and spike processing techniques for development of the Bioinspired Model Development Environment (BMDE, Chapter 3) and the corticostriatal circuit-inspired brain-machine interface control algorithm (B-BMI).

Figures 2.9 through 2.11 show the plot of chronic recordings obtained from one rat using the techniques presented in Chapter 2.1. The data were recorded during two lever choice trials as explained in the Chapter 2.3. The upper plot in each figure shows the (band-pass filtered) neural signal recorded from one channel during left lever press. The plot in the middle demonstrates the spike activity for the right lever press in the same behavioral paradigm. Vertical black lines in these plots indicate the time point for the trial initiation and y-axis demonstrates the signal amplitude. Vertical green and red lines indicate the time of trial termination by a correct lever press. The lowermost graph in each figure shows the firing rate estimate of one of the units isolated from the corresponding recording channel. Firing rate estimates were evaluated using non-overlapping 100 ms bins, based on 25 trials for each target side. The red lines indicate the firing rate estimate for a left lever press and the green lines show the firing rate

estimate for the right lever press. Vertical black lines in these graphs point the trial initiation time and vertical blue lines show the average of the trial termination times evaluated based on all 50 trials.

In Figure 2.11 we can see the strong inhibition for a left lever press (vertical red line) and extreme excitation for the right lever press (vertical green line) for one channel of the recordings. The motor cortex forelimb area was validated through *in vivo* recordings from several rats. The experience acquired from these recordings and behavioral experiments has been significantly useful in the design and the implementation of the BMDE and the bioinspired brain machine interface control algorithm (B-BMI). By bilaterally implanting the presented 8 channel Pt/Ir microwire array (a total of 16 channels) and using the presented surgical techniques, we were able to isolate 10-12 units from the recordings even 8-10 months after surgery. We believe utilization of an automatic micropositioner which is capable of penetrating the microwire arrays very slowly (with a step size of 1-2 μm per second) will boost the microelectrode array yield, the number of isolated units per channel per microelectrode array.

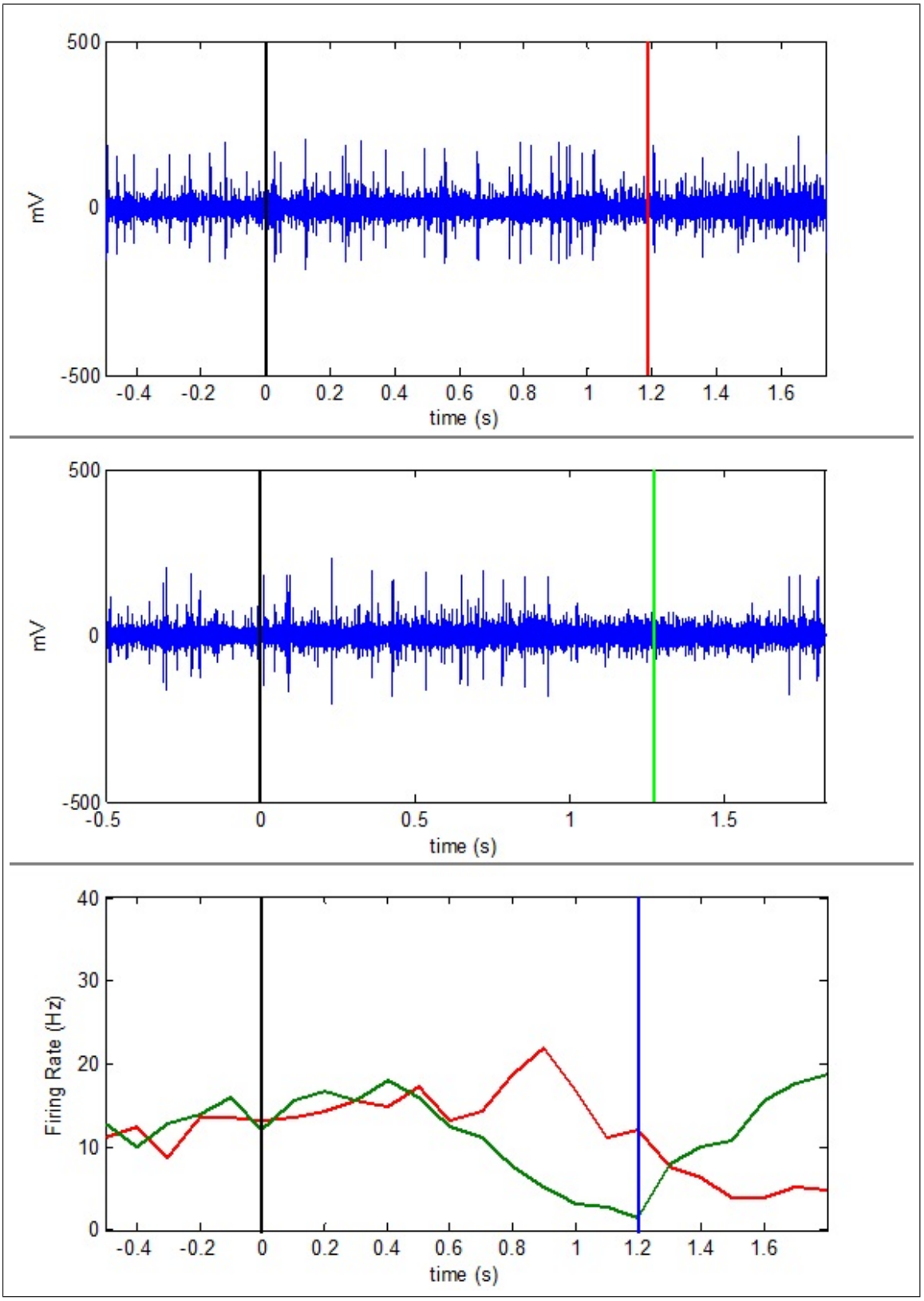

Figure 2.9 Neural recording for a two-lever choice task (excitation for left lever press).

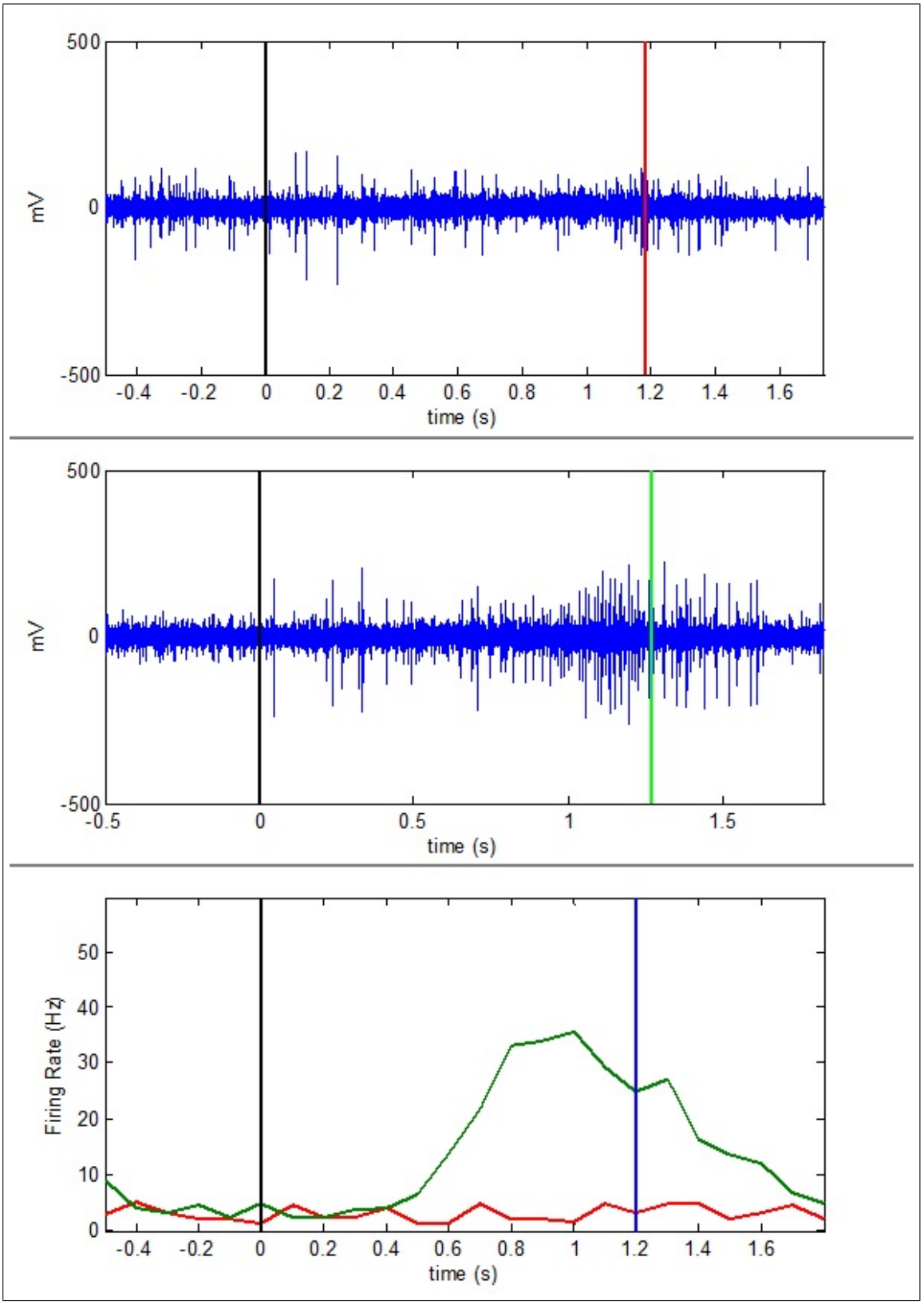

Figure 2.10 Neural recording for a two-lever choice task (excitation for right lever press).

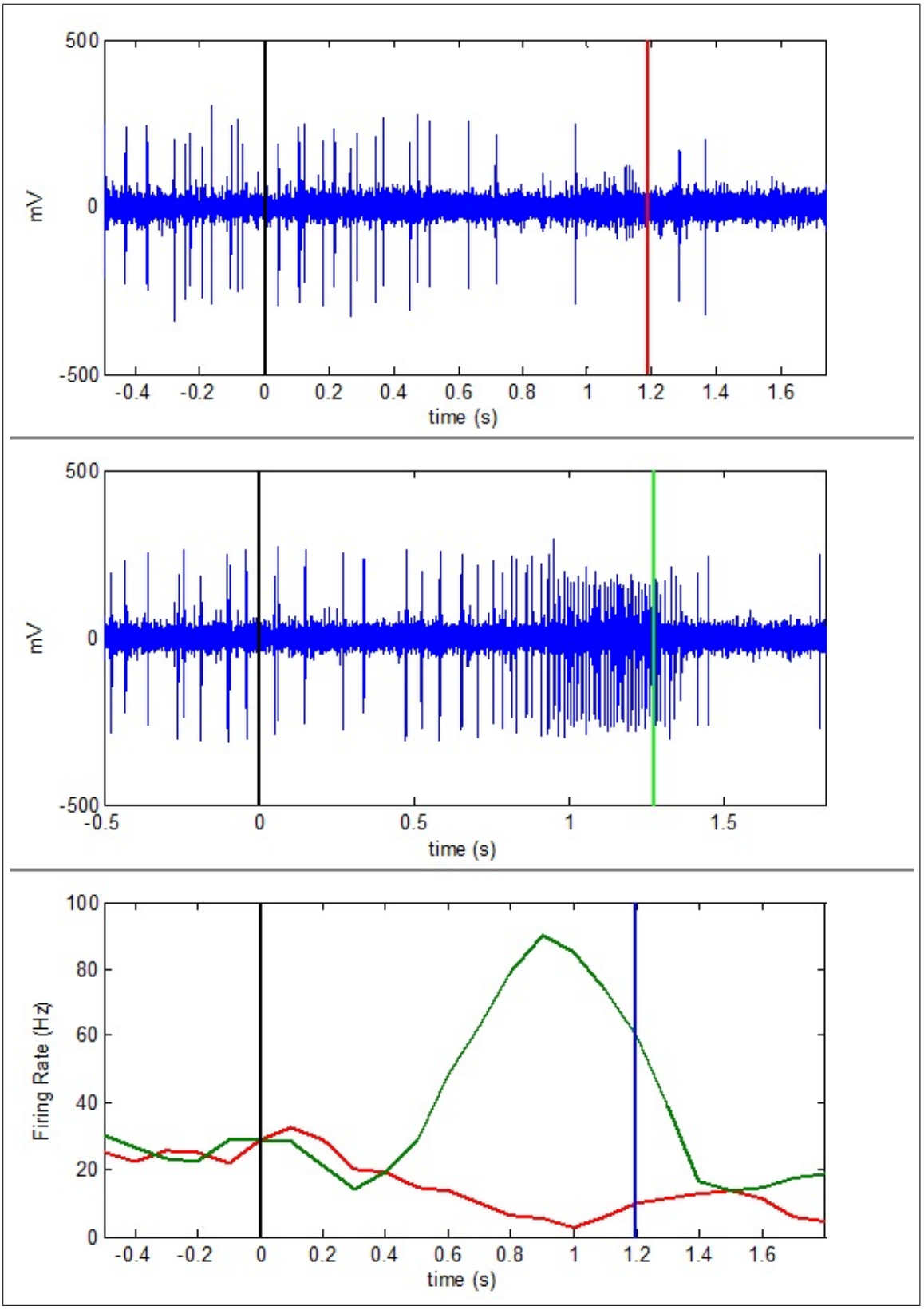

Figure 2.11 Neural recording for a two-lever choice task (excitation for right lever press).

3. BIOINSPIRED MODEL DEVELOPMENT ENVIRONMENT (BMDE)

During the development of the rodent behavioral environment, we simultaneously implemented the hardware and software platforms for conducting the behavioral experiments and the development of neuroprosthetic control algorithms. As mentioned in the motivation section of this dissertation (Chapter 1.2), we focused on the development of neurally-inspired BMI controllers which operate by simulating spiking model neurons. Since this neuroprosthetic control paradigm is fundamentally distinct from those of traditional input-output model-based systems (Figure 1.1), we first needed to design a novel, practical and flexible research platform which enables real-time neural network simulations, establishment of simulated synaptic connections between the real neurons and the model neurons and real-time control of a robotic arm in accordance with the outputs of the simulated neurons (Figure 1.3).

Our research for development of bioinspired BMI control algorithms necessitated not only satisfying the real-time constraints for biological/*in silico* neuronal network interactions but also enabling visualization of dynamics of both real and *in silico* neurons. Here the dynamics for real neurons include their spike waveforms and spiking patterns. The dynamics of *in silico* neurons include membrane potentials, synaptic weights and spiking patterns. In this sense, we sought for computing systems which enable both real-time information processing and execution of graphical user interfaces. Consequently, we decided to utilize a quad-core personal computer (PC) and equip it with the Real-time Application Interface (RTAI, www.rtai.org). We call this PC-based platform 'Bioinspired Model Development Environment (BMDE)'.

3.1 Real-time Application Interface (RTAI)

RTAI is an open-source, hard real-time extension for Linux operating system [40]; it guarantees strict timing constraints of real-time applications while also allowing execution of standard Linux features and services (e.g. window system, keyboard/mouse inputs, file system, Linux applications etc.) in the same system. In order to achieve this, RTAI handles the Linux operating system as a lowest priority task and enables (highest priority) real-time tasks to preempt Linux services whenever needed. As a consequence, no unexpected delays occur in execution of the real-time (RT) tasks and standard Linux services are run only when no RT task is executing in the system Figure 3.1. From the point of the user, working of the system remains the same as in a standard Linux operating system [40].

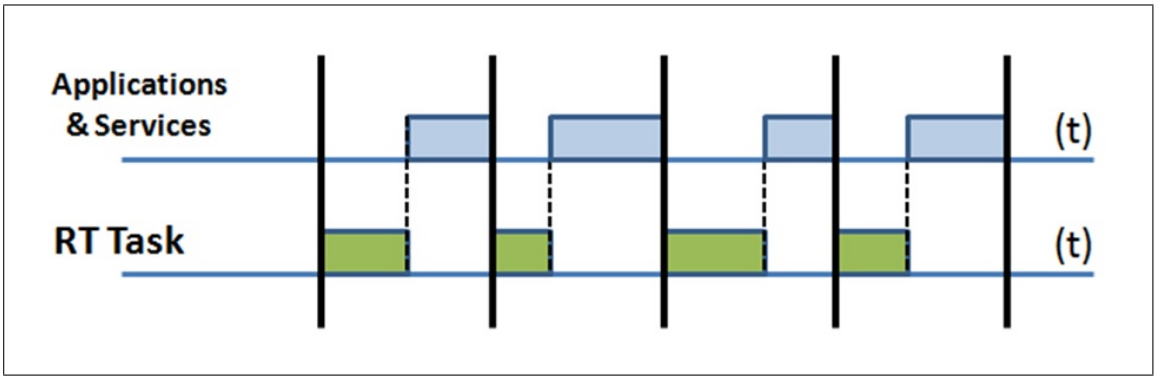

Figure 3.1 Time-share between a RT task and standard Linux services/applications. RT task preempts applications whenever needed. Vertical black lines indicate execution times of the RT task. Green boxes represent the time period in which RT task is run. No unexpected delay or interruption occurs in its execution. Standard Linux services are allowed to run when no RT task is running (shown by blue boxes).

Based on these features of RTAI, we developed the BMDE to provide an all-in-one solution for *in vivo* neuronal modeling studies; it is equipped with both RT tasks and non-time critical applications. While the RT tasks guarantee the timing constraints for biological-*in silico* neuronal interactions through simulated synapses and real-time SNN simulations, the non-time critical applications enable live visualization of experimental data and execution of graphical user interfaces (GUIs) for management of the behavioral experiments. Thus, the experimenter is able to monitor the spiking activity patterns and dynamics of the simulated neurons online while the RT tasks are

performing the time critical operations in the background.

3.2 Generic Software Architecture of the BMDE

The design philosophy of the BMDE aims to combine the flexibility of software-based real-time signal processors and SNN simulators with powerful hardware resources around a standalone personal computer. As illustrated in Figure 3.2, the BMDE mainly executes five RT tasks which present a framework for implementation of bioinspired, SNN-based control algorithms and behavioral paradigms for *in vivo* modeling studies. These tasks are: 1) spike sorting, 2) SNN simulation, 3) prosthetic control, 4) digital input-output (DIO) control and 5) experiment control tasks. In this section, we briefly explain the role of these tasks; more detailed explanation about how each task works will be given in Chapter 5 by demonstrating the implementation of a corticostriatal circuit-inspired brain-machine interface control algorithm (B-BMI).

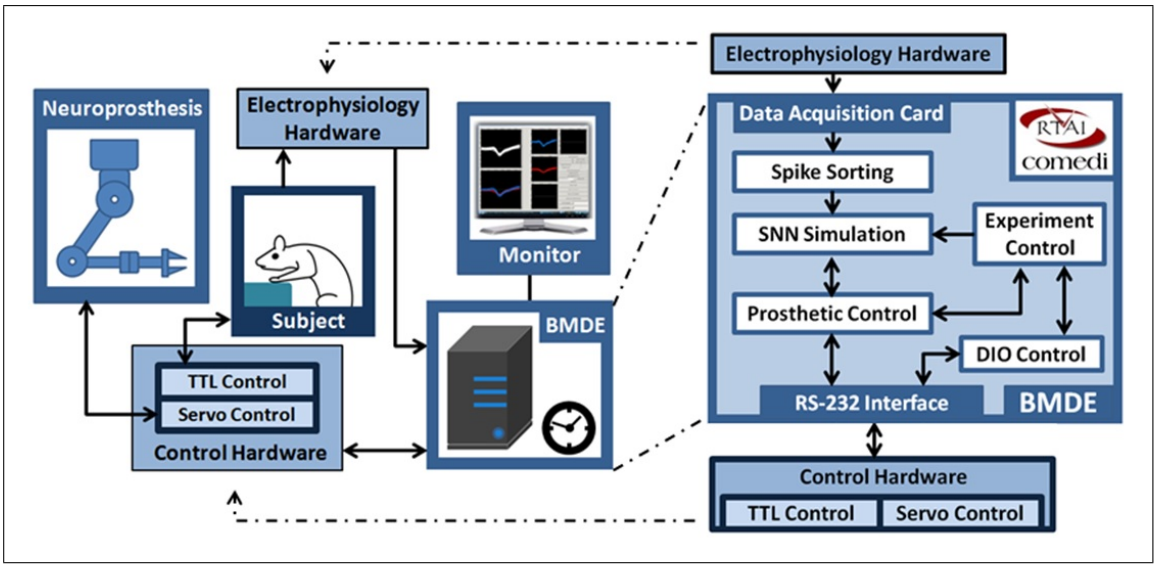

Figure 3.2 Generic real-time tasks and hardware components of the BMDE. The BMDE is implemented around a desktop PC. The real-time tasks are spike sorting, SNN simulation, prosthetic control, digital input-output (DIO) control and experiment control tasks. Electrophysiology hardware provides analog neural signal inputs to the data acquisition board. The prosthetic and digital input-output (DIO) control tasks communicate with the control hardware through an RS-232 interface card. The direction of flow of communication between real-time tasks are shown using arrows. The formats and contents of the messages are given in Appendix A.

3.2.1 Spike Sorting Task

The spike sorting task of the BMDE works in conjunction with a data acquisition (DAQ) device to acquire and process the analog neural signals provided by a standard extracellular neural recording system consisting of microelectrode assemblies, signal filters and amplifiers Figure 2.1. In the BMDE, the DAQ device (National Instruments, PCIe-6259) is configured to perform continuous analog-to-digital conversion (ADC) with a sampling rate of 31.25 KHz per channel and the spike sorting task executes the signal processing routines for extracting single-unit spikes through the neural data provided by the DAQ device. For each detected single-unit spike, the spike sorting task delivers an event to the postsynaptic neurons of the SNN simulation task of the BMDE to provide simulated synaptic interactions between biological and *in silico* neurons.

In order to extract the single unit spikes from the recordings, the spike sorting task filters the neural signals by a 4th order Butterworth digital band-pass filter (cut-off frequency = 400Hz-8KHz), up-samples the filtered neural data to 62.5 KHz by cubic interpolation to improve spike alignment performance, detects the neural spikes by level thresholding and performs spike sorting by Gaussian template matching [41]. The thresholds and templates for spike detection and sorting are manually determined by the experimenter using the spike sorting GUI of the BMDE (Figure 3.3). The single-unit spike templates are formed using the waveform of at least sixty detected spikes, each of which is represented by eighteen data points. In the BMDE, each single-unit which is isolated from a recording channel is modeled with a multivariate Gaussian distribution, $N(\mu,\Sigma)$. The likelihood of the each detected spike given a particular single-unit or class C_i is [42]:

$$p(x|C_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} exp\Big[-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)\Big] \tag{3.1}$$

where x is the d-dimensional spike data vector (d=18), μ_i and Σ_i are mean and covariance matrix for unit C_i, respecitvely. The detected spikes are classifed into up to three units.

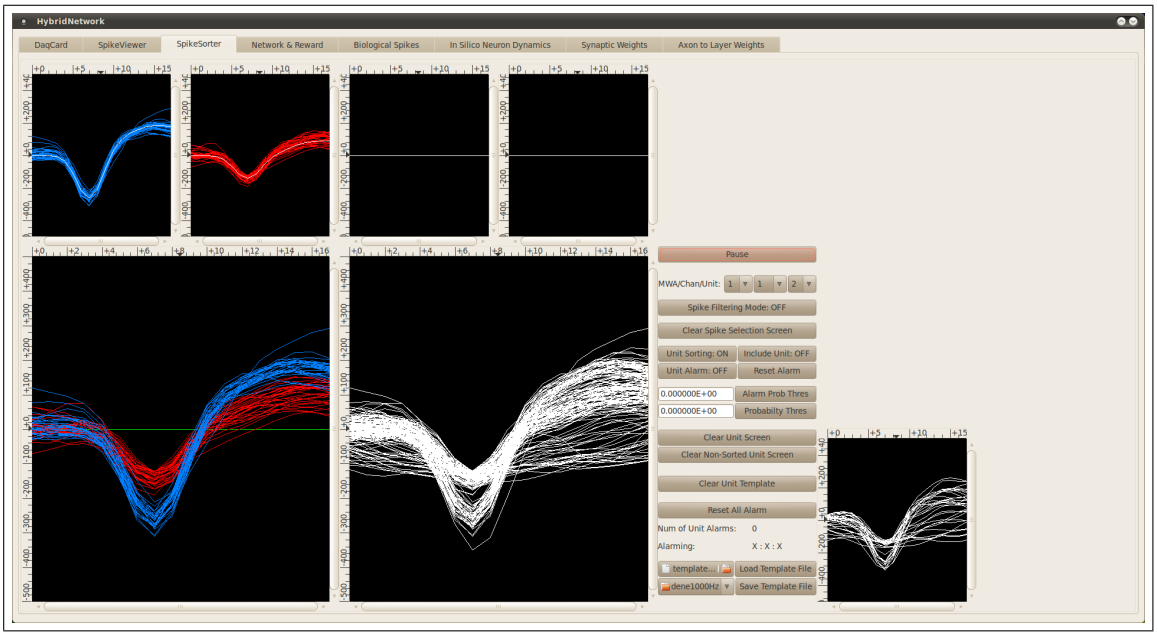

Figure 3.3 Snapshot from the spike sorting GUI during *in vivo* recording.

3.2.2 SNN Simulation Task

The SNN simulation task executes every 2 ms and performs numerical integrations to evaluate the dynamics of the neurons in the SNN. In the present implementation of the BMDE, there are two SNN simulation tasks each of which are assigned to a different CPU core (Table 3.1) and simulates half the neurons in the SNN. Thus, it becomes possible to allocate two cores of the CPU for computationally intensive SNN simulations. For implementation of the SNN, we utilize Izhikevich's simple neuron model, which is capable of exhibiting the rich dynamic repertoire of real neurons with simple differential equations [43, 44, 45]. For numerical integrations for simulation of the SNN, we utilize the Parker-Sochacki integration method [46, 47]. The method provides double precision in the numerical integrations with a simulation time applicable in the BMDE. The pseudo-code for the routines of SNN simulation task (and other real-time tasks of the BMDE) is located in Appendix B section of this dissertation.

3.2.3 Prosthetic Control Task

The prosthetic control and the DIO (digital input-output) control tasks operate in cooperation with the in-house built, microcontroller-based (Microchip, PIC18F4520) control hardware, which enables the control of the experimental environment components by its embedded software modules (i.e. the servo control module and the TTL (Transistor-Transistor Logic) control module). To achieve the communication with the control hardware, the BMDE utilizes a commercially available generic RS-232 interface controller which transmits and receives data at 115.2 Kbaud per second.

The prosthetic control task is the intermediary between SNN simulation tasks and the servo control module of the control hardware. It buffers the spike events received from the output layer of the SNN and translates them into pulse width commands to be handled by the servo control module. The servo control module then drives the three actuators or joints of a customized version of Lynxmotion AL5D robotic arm (Swanton, VT, Figure 3.4) and returns the angle values of the joints through the same interface. By receiving the joint angles of the robotic arm, the prosthetic control task calculates the Cartesian position of the tip of the arm by forward kinematics [48].

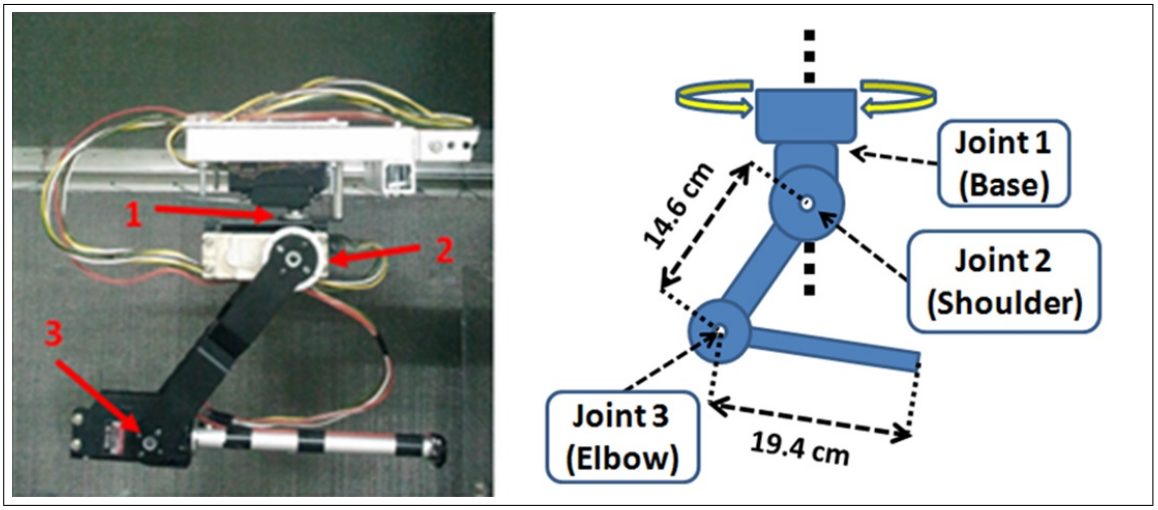

Figure 3.4 Side view (left) of the customized Lynxmotion robotic arm and its dimensions (right). The red arrows in the figure indicates the location of the joints of the robotic arm.

3.2.4 Digital Input-Output (DIO) Control Task

The DIO control task receives the status of the digital inputs from experimental environment through the TTL control module of the control hardware and determines if the time to trigger an event is expired for a digital input. For instance, if a lever is pressed for a certain amount of time, it can trigger an event (successful lever press) and send it to experiment control task to request a trial initiation. In addition, it delivers digital outputs to the TTL control module to alter the binary state of the experimental environment components (e.g., turn LED off, release reward etc.).

3.2.5 Experiment control task

The experiment control task is the management center of the behavioral experiments. By receiving messages from the prosthetic and DIO control tasks, it decides if a trial should be initiated, ended, rewarded etc. In addition, it informs the downstream tasks (i.e., SNN simulation, DIO control and prosthetic control tasks) about the decisions it made so that they take action to apply the requirements of the experimental paradigm.

3.3 Assigning Real-time Tasks to Specific CPU Cores

To utilize the system resources efficiently, each RT task of the BMDE is assigned to run on a particular core of the CPU (Intel i7-950, Table 1). These tasks communicate with each other using the shared memory feature of the RTAI and the messaging libraries developed in this work.

Table 3.1
The RT tasks of the BMDE and CPU core assignments.

RT Task	CPU Core	Task Period
Spike Sorting	0	512 μs
SNN Simulation	1 & 2	2 ms
Prosthetic Control	3	2 ms
Experiment Control	3	2 ms
DIO Control	3	2 ms

3.4 Data Visualization and Recording in the BMDE

In order to facilitate the SNN-based BMI control studies, the BMDE was also equipped with non-time critical applications to provide GUIs and online data visualization tools for enabling the management process of the experiments. For the implementation of the GUIs, we utilized open-source GTK+ libraries (www.gtk.org). In addition, using the GtkDatabox libraries (sourceforge.net/projects/gtkdatabox), we implemented online data visualization utilities which provide live display of the continuously changing signals (e.g. neural signals, dynamics of *in silico* neurons, neuronal spike trains etc.).

The GUI for the spike sorting process enables the configuration of the DAQ device and visualization of the acquired neural signals through a software-based oscilloscope. It also provides visualization tools for determination of the thresholds and templates for spike sorting process (Figure 3.3). The GUI related to the SNN simulation allows the user to build a neural network consisting of spiking neurons and visualize the dynamics of the neurons and spiking activity patterns of the real neurons during the behavioral experiments. The other GUIs provide the software forms for adjustment of the parameters related to the management of the experiments (e.g. for submitting the maximum duration of a trial, the length of a valid lever press, etc.) and handle online data recording utilities which periodically save experimental data to the hard drive of the system.

3.5 Real-time Neuronal Interaction in the Hybrid Neural Network

To provide biologically realistic synaptic interactions between real and *in silico* neurons in the bioinspired neuroprosthetic control paradigm, the synaptic delays between these neurons need to be deterministic and in the order of several milliseconds. In other words, excessively high and continuously varying delays in transmission of spike events between real and simulated neurons might disrupt the information encoded by timing of the spikes in the hybrid biological/*in silico* neural network and reduce the performance reliability and reproducibility of the BMI controller. In this context, it is crucial to deterministically extract the time of the single-unit spikes from the neural recordings and deliver them to the post-synaptic *in silico* targets with consistent synaptic delays. To this end, in implementation of the BMDE, we utilized the open-source DAQ card drivers and application programming interfaces (APIs) provided by the COMEDI (Linux Control and Measurement Device Interface, www.comedi.org) project and the DAQ-related APIs provided by the RTAI developers (www.rtai.org). Using these drivers and APIs, it is possible to bypass the interrupt management layers of the standard Linux kernel [40] and provide deterministic responses to the hardware interrupts of the DAQ device; a high priority RT task can be immediately executed whenever the DAQ device generates an interrupt subsequent to acquiring preconfigured amount of samples. Thence, in the case of recording of extracellular neural signals, the executed RT task can deterministically extract the time of the spikes through the acquired signals.

In the BMDE, the 32-channel DAQ board is configured to perform continuous ADC at a sampling rate of 31.25 KHz per channel and deliver all on-board buffered data to the system memory whenever 16 samples for each channel are acquired; the period for delivery of the data to system memory corresponds to 512 μs. After transmission of the data to the system memory, the DAQ device generates a hardware interrupt and the spike sorting task is executed in response to this interrupt to detect the single-unit spikes through the recordings. To extract the single-unit spikes, it 1) filters the neural

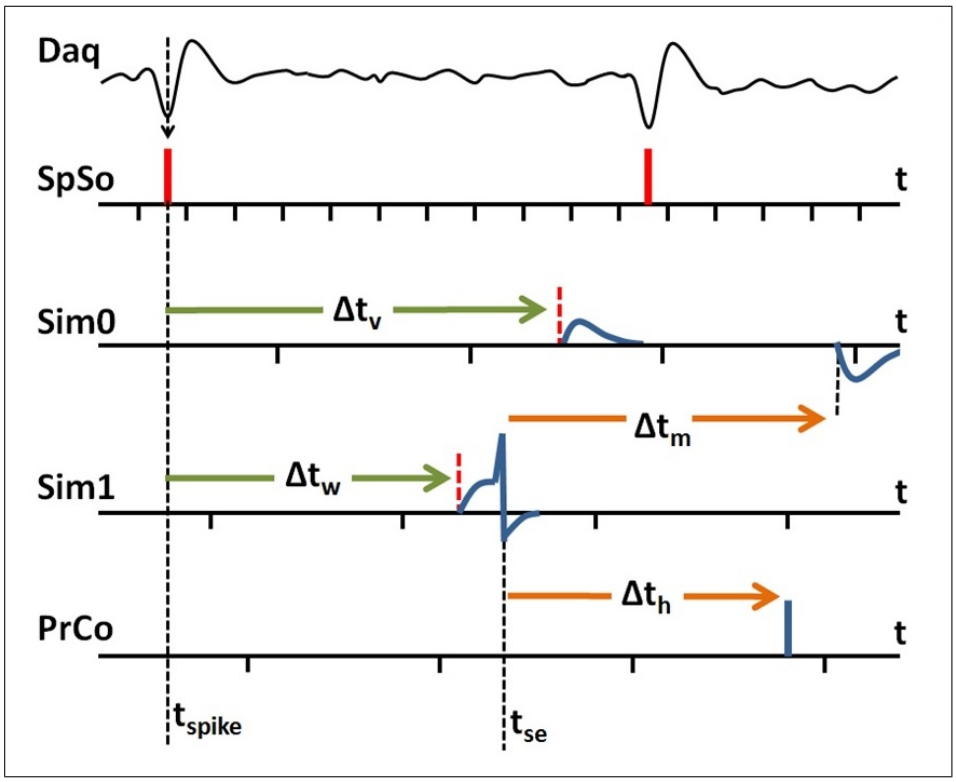

Figure 3.5 An example of spike event delivery between the tasks of the BMDE. SpSo: spike sorting task, Sim0 & Sim1: SNN simulation tasks, PrCo: prosthetic control task. The execution times of tasks are shown by the ticks on time axes. The tasks are periodically triggered. Task period for SpSo is 512μs, and 2 ms for Sim0, Sim1 and PrCo as shown in Table 3.1. Their executions do not have to be synchronous. In the present example, a neural spike time (t_{spike}) is determined according to the lowest peak of the spike waveform through the extracellular recordings. A synaptic event for this spike is scheduled and delivered to the post-synaptic *in silico* neurons to be handled at t_{spike}+Δt_v and t_{spike}+Δt_w. After being depolarized, the neuron simulated by Sim1, but not the one simulated by Sim0, generates a spike at t_{se} and then events are scheduled for t_{se}+Δt_m and t_{se}+Δt_h to be handled by Sim0 and PrCo, respectively. Since the neuron simulated by Sim1 is inhibitory, it leads to hyperpolarization in the post-synaptic neuron simulated by Sim0.

signals by a 4th order Butterworth digital band-pass filter (cut-off frequency = 400Hz-8KHz), 2) up-samples the filtered neural data to 62.5 KHz by cubic interpolation to improve spike alignment performance, 3) detects the neural spikes by level thresholding and 4) performs spike sorting by Gaussian template matching [41]. After performing these steps, the spike sorting task finally timestamps the detected single-unit spikes according to the lowest peak of their waveform and schedules synaptic events for the post-synaptic *in silico* targets (Figure 3.5). The SNN simulation tasks then sort the incoming synaptic events in the time domain and sequentially handle these events to evaluate the dynamics of the *in silico* cells. Consequently, biologically realistic synaptic interactions between real and *in silico* neurons are realized in the BMDE. More details

about the methods for simulating the SNN will be given in Chapter 5.

3.6 The Control Hardware

The control hardware (Figure 3.2) was developed around a microcontroller (Microchip, PIC18F4520). The schematic diagram for the control hardware is given in Appendix C. Its embedded software code was written in C programming language (pseudocode is given in Appendix D) and compiled using Microchip-C18 compiler. In the development process of the control hardware, we utilized the Microchip In-Circuit Debugger (ICD) 2. The communication between the BMDE and the microcontroller is realized through the RS-232 standard with a data transmission rate of 115.2 Kbaud per second.

In order to perform the control of the robotic arm and the experimental environment components, the BMDE delivers the 10-byte message (shown in Figure 3.6) to the control hardware every 26 ms.

'P'	CMD	PW_LB	PW_HB	PW_LB	PW_HB	PW_LB	PW_HB	0xFF	0xFF
INIT_MSG	TTL_CMD	SERVO_0		SERVO_1		SERVO_2		EOM	EOM

Figure 3.6 The format of the message delivered by the BMDE to the control hardware.

The message starts with a 'P' character to inform the control hardware that a message delivery by the BMDE is initiated (INIT_MSG) and two-byte End of Message (EOM) indicates that the message delivery is complete. These bytes are also used for error check in the messaging between the BMDE and the control hardware. The byte corresponding to TTL_CMD is prepared by the DIO control task of the BMDE and includes the commands to alter the binary state of the experimental environment components. Each bit of the TTL_CMD message corresponds to the binary state of one of the experimental environment components. According to the status of the bits of the TTL_CMD, the TTL control module of the control hardware, for instance, turns

the LEDs on/off or releases a water reward through controlling a solenoid valve etc.

The 2-byte pulse width commands (SERVO_0, SERVO_1 and SERVO_2) are prepared by the prosthetic control task of the BMDE. The servo control module of the control hardware handles these messages to manipulate the Lynxmotion robotic arm by delivering pulse width signals to its servo motors. The pulse widths for servo motor control are accurately applied using the timer interrupt services of the microcontroller (MCU).

Transmission of the 10-byte message shown in Figure 3.6 to the control hardware triggers the execution of the code related to servo control and TTL control modules of the control hardware. In other words, when there is no message from the BMDE, the control hardware stays idle and does not process any output related to the robotic arm or (binary-stated) experiment environment components. When a message is received, it is processed by the modules of the control hardware as explained in the previous paragraphs. When the message processing is complete, 1) the servo control module measures the joint angles of the servo motors of the robotic arm using analog-to-digital converter (ADC) module of the microcontroller (MCU), 2) the TTL control module monitors the input pins (pin 34 and 35 shown in the schematic diagram in Appendix C) of the MCU to detect the binary status of experimental environment components such as cage levers or infrared beam sensors. Whenever these processes are complete, the control hardware transmits a 10-byte message (shown in Figure 3.7) to the BMDE through the RS-232 interface.

'P'	CMD	PO_LB	PO_HB	PO_LB	PO_HB	PO_LB	PO_HB	0xFF	0xFF
INIT_MSG	TTL_STATUS	SERVO_0		SERVO_1		SERVO_2		EOM	EOM

Figure 3.7 The format of the message from the control hardware to the BMDE.

The message starts with a 'P' character to inform the control hardware that a message delivery by the BMDE is initiated (INIT_MSG) and two-byte End of Message (EOM) indicates that the message delivery is complete. The byte corresponding to TTL_STATUS includes the binary state of the experimental environment components

and is handled by the DIO control task of the BMDE. The 2-byte data corresponding to the joint angles of the robotic arm (SERVO_0, SERVO_1 and SERVO_2) are processes by the prosthetic control task of the BMDE to evaluate the position of the tip of the robotic arm.

4. BMI CONTROLLER INSPIRED BY THE CORTICOSTRIATAL CIRCUIT

The BMDE described in this dissertation has been developed to enable implementation of biologically plausible neuroprosthetic systems which perform the control of a robotic arm by simulating the neuronal dynamics of the motor system circuits (Figure 4.1). On the way to designing large scale models of circuits of the motor system, it would be logical to start by generating a BMI control algorithm which is based on a limited number of spiking neurons and is inspired by information processing principles in motor-related circuits. Thus, it would be possible to build more complex, larger scale models as our understanding of information processing principles in the neural circuits improves using neurally-inspired, SNN-based BMI controllers in *in vivo* neuroprosthetic control experiments. In this context, we are interested in developing a BMI control algorithm which is inspired by the motor corticostriatal circuit since 1) this circuit plays a key role in motor performance, motor learning and reinforcement learning [49, 50], and 2) the striatum, as the input structure of the basal ganglia circuit, is considered to have a prominent role in action or motor program selection [51]. In this section, a brief overview of motor system is presented, the neurophysiology of the corticostriatal circuit (CSC) is described and a SNN-based BMI controller is introduced for one-dimensional control of a robotic arm.

4.1 Motor System

Major brain regions involved in the motor system are the cerebral cortex, the basal ganglia, the cerebellum, the brain stem and the spinal cord. Since most neuroprosthetic systems are designed to mimic voluntary movements (rather than involuntary movements) through a robotic arm, it is convenient to focus on brain structures involved in the preparation, planning and execution of motor behaviors. For a voluntary movement, the command signals originate in the association areas of the cerebral cortex. The movement, subsequently, is planned in the cortex, the basal ganglia and

the lateral portions of the cerebellum; electrophysiological recordings show that these areas are activated prior to execution of the movement [52]. The motor cortex, later on, executes movements through the corticospinal tract. The cerebellum and the basal ganglia, receiving inputs from the cortex, modulate motor cortical activity through the loops involving the thalamus and the brain stem, without any direct output to the spinal cord (Figure 4.1). Instead, these structures significantly influence the processing of motor control by modulating the output of the pathways ending in the spinal cord, such as the cortex and the brain stem [53].

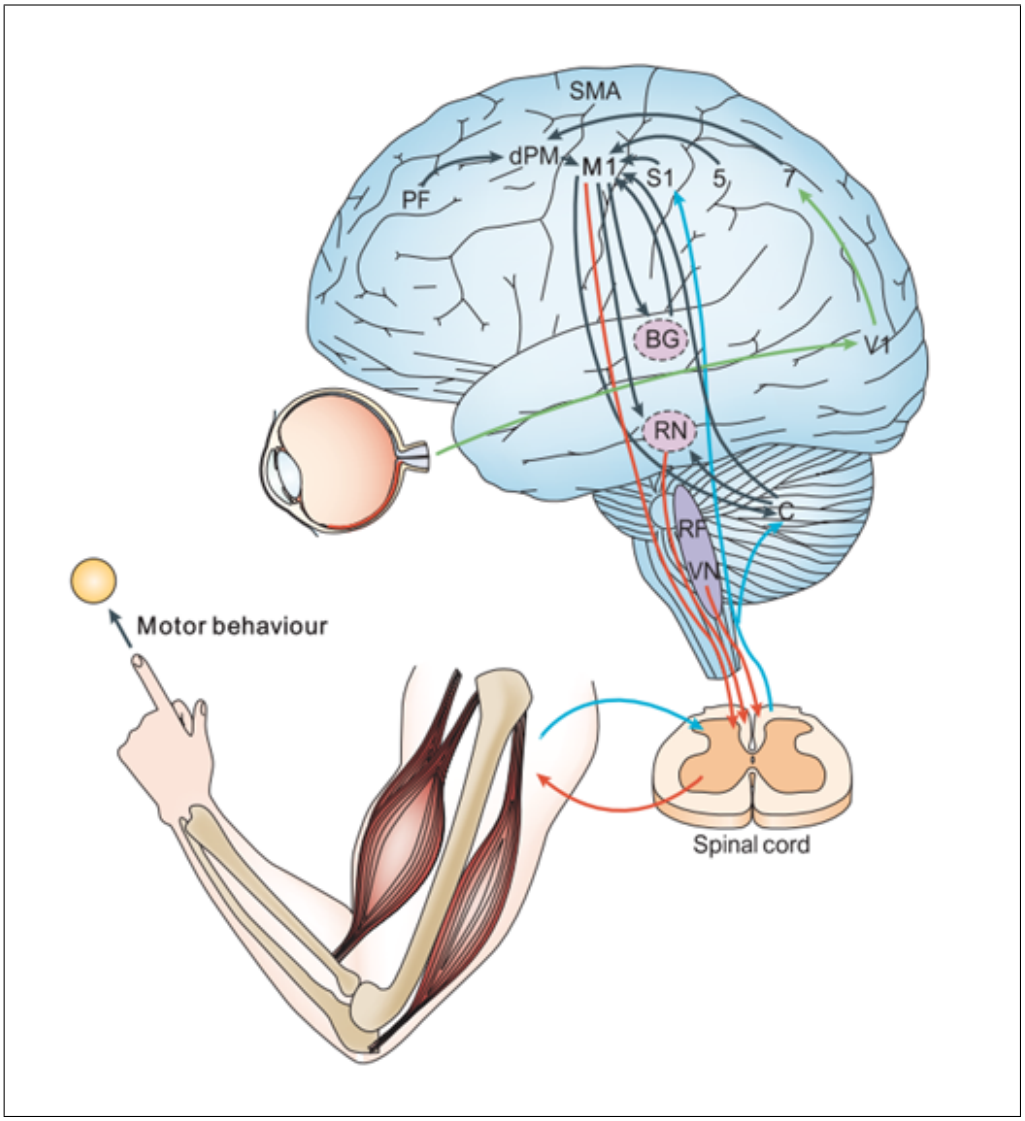

Figure 4.1 The motor system components (RF: reticular formation, VN: vestibular nuclei, M1: primary motor cortex, S1: primary somatosensory cortex, 5: parietal cortex area 5, BG: basal ganglia, C: cerebellum, RN: red nucleus, V1:primary visual cortex, 7: region of posterior parietal cortex, dPM: dorsal premotor cortex, SMA: supplementary motor area, PF: prefrontal cortex, adapted from [54]).

The spinal cord receives inputs from the motor cortical areas (i.e. primary, supplementary and premotor cortex) for execution of voluntary movement. Other cortical areas (i.e. parietal lobe and the primary somatosensory area) also project to the spinal cord (40% of all corticospinal axons in the human). In addition, other projections to spinal cord originate from the brain stem structures (i.e. vestibular nuclei, reticular formation, red nucleus and superior colliculus) and are related to posture and motor coordination [52].

4.2 Basal Ganglia System

The basal ganglia comprise four nuclei and play a major role in voluntary movement through the complex anatomical loops, which involve the thalamus and the cerebral cortex (Figure 4.2). The four nuclei of the basal ganglia system are: the striatum, the globus pallidus, the subthalamic nucleus and the substantia nigra. The globus pallidus is composed of two compartments, namely pallidus interna and externa, and these compartments have different functions in the basal ganglia system. In addition, substantia nigra also consists of substantia nigra pars reticulata and substantia nigra pars compacta. Substantia nigra pars compacta includes dopaminergic neurons projecting to striatum modulating motor activity and synaptic plasticity.

The output centers of the basal ganglia (i.e. globus pallidus interna (GPi) and substantia nigra pars reticulata (SNr)) are tonically active and continuously inhibit the thalamus. In this context, basal ganglia appear to apply a powerful brake on voluntary movements. This inhibition is selectively disinhibited by projections from the striatum. Therefore, the striatum, entry point of the basal ganglia circuitry, has a prominent role about which action or motor program should be executed [51].

The striatum is the largest component of the basal ganglia circuitry. It is mainly composed of GABAergic medium spiny neurons (MSNs, approximately 95% in the rat [55]). The MSNs primarily receive excitatory synaptic inputs from the cortical areas and the thalamus (80% of all synapses in the striatum), and project their axons to

the other nuclei of the basal ganglia [56]. The collateral branches of these axons, additionally, form inhibitory connections between these projection neurons. Other inhibitory synapses to the MSNs come from the other nuclei of the basal ganglia, local cholinergic and GABAergic interneurons [57].

The striatal MSNs are separated into two types according to the relative proportions of D1- and D2-type dopamine receptors they express. These subpopulations of MSNs project to different nuclei of the basal ganglia [58]:

D1-type MSNs prominently deliver their axons to 1) substantia nigra pars reticulata (SNr) and 2) to globus pallidus interna (GPi, not shown in Figure 4.2). The neurons of the SNr and GPi, which are mainly inhibitory, then project to the thalamus.

D2-type MSNs send their axons to globus pallidus externa (GPe), which has an inhibitory effect on subthalamic nucleus (STN) and the STN neurons excite SNr through glutamatergic synapses.

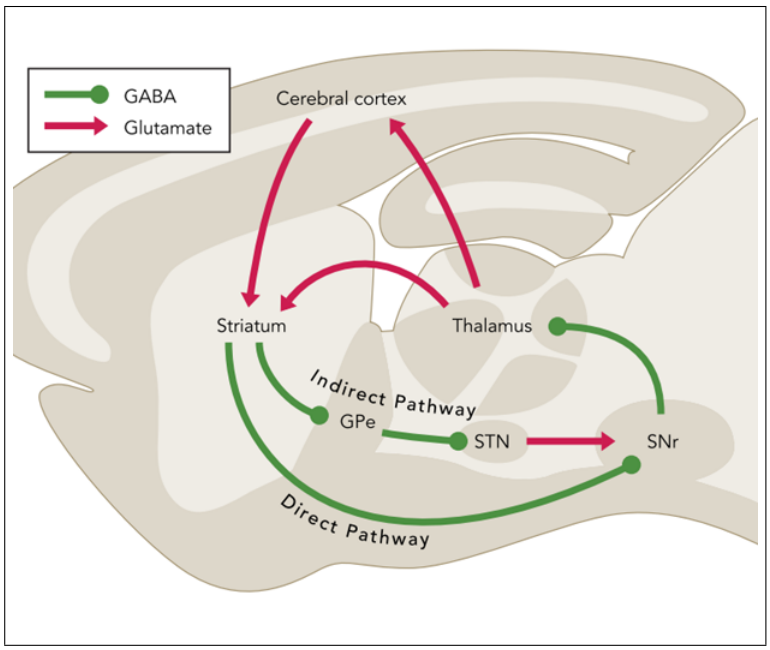

Figure 4.2 Major basal ganglia system interconnections in the rodent brain (GPe: external globus pallidus, STN: subthalamic nucleus, SNr: substantia nigra pars reticulata, from [59]).

The projections of these two different types of MSNs (D1 and D2 type MSNs) modulate the cortical activity through antagonistically working direct and indirect pathways as shown in Figure 4.2. In the presence of dopamine, the excitability of the D1-type MSNs increases, and the net excitatory effect of the direct pathway on the cerebral cortex is augmented. In contrast, the presence of dopamine inhibits D2-type MSNs and reduces the net inhibitory effect of the indirect pathway on the cerebral cortex. In this scenario, the net activity in the cerebral cortex increases and the motor behavior is promoted through the corticospinal and the corticobulbar tracts. On the contrary, the absence of dopamine in the striatum enhances the inhibitory effect of the indirect pathway on the cortex and leads to attenuated movement. In this sense, dopamine maintains the balance between the direct and indirect pathway activation in the cortico-basal ganglia-thalamic system and plays a key role in motor control. The substantia nigra pars compacta (SNc) is the midbrain structures involved in the basal ganglia system and providing dopaminergic inputs to the MSNs of the striatum. The loss of dopaminergic neurons in this structure leads to pathologies such as Parkinson's disease [60].

4.3 Corticostriatal Plasticity

Corticostriatal plasticity plays a critical role in motor [49, 61] and neuroprosthetic skill learning [31]. In addition to its excitatory and inhibitory effects on striatal MSNs, dopaminergic inputs from the substantia nigra modulate the synaptic plasticity in the corticostriatal synapses [62, 63]. In fact, the presence of dopamine is essential for occurrence of spike-timing-dependent plasticity [64], including both long-term potentiation (LTP) and long-term depression (LTD). Such requirement for synaptic plasticity in the corticostriatal pathway is characterized by the 'three-factor rule': For induction of LTP, bursts of dopaminergic neurons, correlated presynaptic and postsynaptic activity is necessary. On the contrary, low levels of dopamine combined with correlated pre- and postsynaptic activity can leads to LTD [62, 65].

The phasic changes in dopaminergic neuronal activity, which may lead to LTP

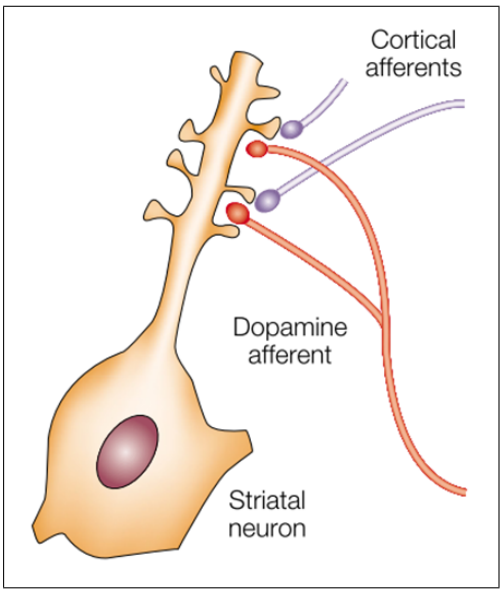

Figure 4.3 Glutamatergic inputs from the cortex and dopaminergic inputs from substantia nigra pars compacta or ventral tegmental area to medium spiny neurons (MSNs) of the striatum. Dopaminergic inputs modulate the plasticity in the glutamatergic synapses [66].

and LTD in the corticostriatal synapses [62, 67], was presented by Shultz and colleague's experiments several decades ago [68, 69]. These phasic changes were shown to encode the difference between the reward predicted and the actual reward received, namely reward prediction error. If a reward is unexpectedly delivered, then the activity of the dopaminergic neurons increases phasically (positive reward prediction error). In contrast, when an expected reward is not delivered, then the tonic activity of dopaminergic neurons phasically decreases (negative reward prediction error). Such phasic changes in dopamine levels in the striatum may modulate the synaptic plasticity during motor learning tasks [49].

In the context of motor performance, some dopaminergic neurons in substantia nigra pars compacta (SNc) increase their activities during execution of arm movements. However, these neurons do not present such response when a reward is delivered. In contrast, the activity pattern of some other dopaminergic neurons in the SNc is not related to motor movements. Instead, these neurons respond to reward delivery [70, 71] by phasic increase in their activities. Further research is required to improve the understanding of how such differences in activity patterns of dopaminergic neurons are integrated into motor performance and learning [49, 72].

In addition to the significant contributions to the scientific knowledge related to cellular mechanisms of corticostriatal plasticity and reward signaling property of dopaminergic neurons, Koralek et al. has recently showed in rodents that the involvement of striatum and occurrence of corticostriatal plasticity is essential in neuroprosthetic learning [31]. During learning the control of a neuroprosthesis, strong relations between the activities of primary motor cortical and striatal neurons develop, and striatal neurons exhibit strong target-related firing rate modulations including both excitation and inhibition in late phase of learning. In addition, disrupting the development of plasticity in the corticostriatal pathway impairs learning neuroprosthetic skills [31]. These findings demonstrate that the striatum is involved in learning and performing not only physical skills [61] but also neuroprosthetic skills.

4.4 Membrane Properties of Medium Spiny Neurons

The medium spiny neurons (MSNs) have a threshold for activation and very low activity profile during resting conditions. They express inward-rectifier K+ (Kir) conductances, which remain open at hyperpolarized membrane potentials but close if the neuron becomes depolarized [51]. Therefore, MSNs present a prominent bistable behavior: MSNs tend to stay in a silent hyperpolarized state ('down-state') as long as they do not receive a strong excitatory input. However, when they are excited, K+ current is deactivated and they remain in depolarized 'up-state', which is close to firing threshold, for a prolonged period [44].

4.5 The Bioinspired BMI Controller

Based on the neuroscientific knowledge related to bistable membrane potential properties of the striatal MSNs and dopamine-dependent plasticity of corticostriatal synapses [51, 73], we generated the Bioinspired BMI controller (B-BMI) to realize a two-target center-out reaching task in one dimensional space (Figures 4.5 & 4.6).

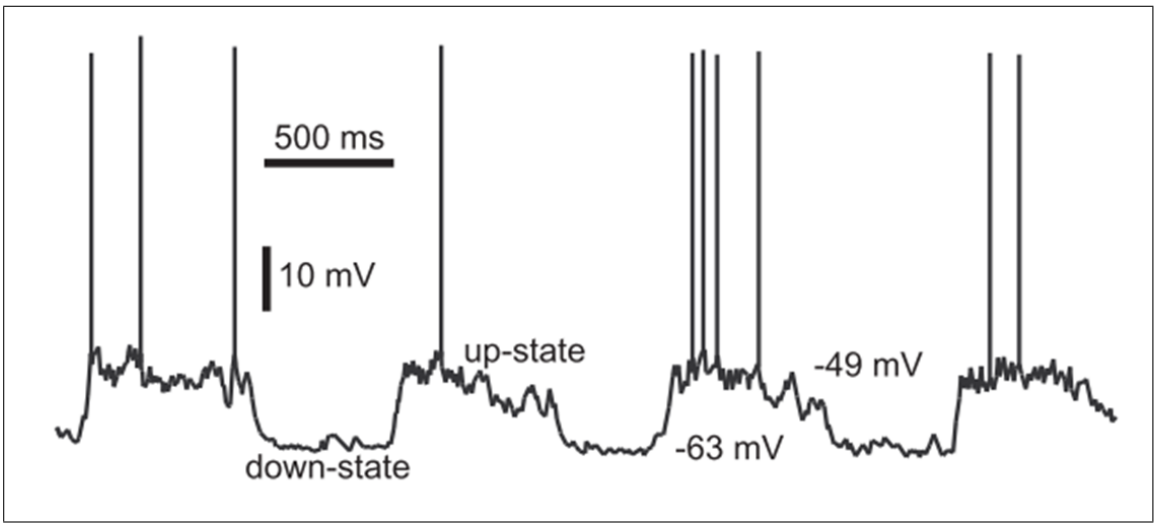

Figure 4.4 Two-state membrane potential dynamics of MSNs *in vivo* [44].

In the B-BMI, the living (extracellularly recorded) motor cortex units are partially connected to the *in silico* MSNs through simulated excitatory synapses. The MSNs are then reciprocally connected through strong lateral inhibitory synapses to build a mechanism for winner-take-all (WTA) competition, a form of which has often been used in computational models of basal ganglia to characterize neural information processing in the striatum [74, 75, 76].

In the present control paradigm, the MSN with the highest spike count is selected as the winning neuron and the prosthetic action (moving to the left or right) corresponding to that neuron is applied by the base servomotor of the robotic arm in one-dimensional space. The spike counts of the MSNs are calculated by binning the generated spikes every 26 milliseconds with a sliding 104 ms time window. In case of equality among the spike count of the MSNs, no winning neuron is selected and no prosthetic action is taken by base servomotor of the robotic arm (remain stationary).

The model MSNs of the BMI controller are described using the equations of Izhikevich's simple neuron model [44]. The synaptic interactions are provided by fast conductance-based synaptic currents [77] as in [47]:

$$Cv' = (v - v_r)(v - v_t) - u - \eta(v - E_\eta) - \gamma(v - E_\gamma) \tag{4.1}$$

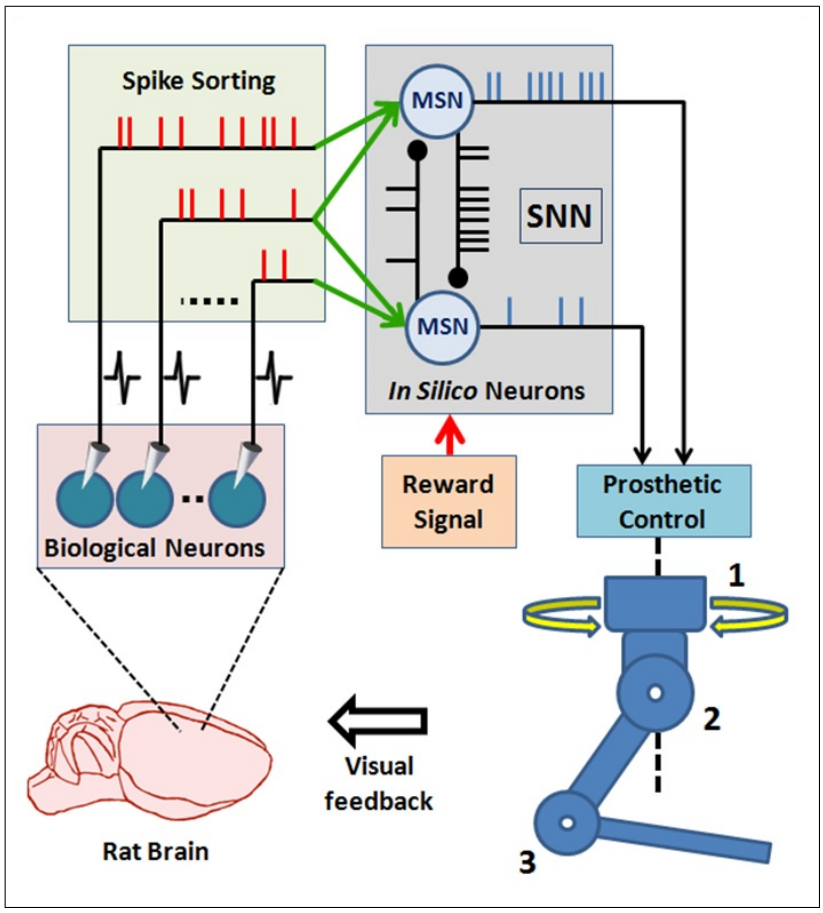

Figure 4.5 Control architecture of the B-BMI which implements a model of synaptic interactions between motor cortex and striatal medium spiny neurons (MSNs). The SNN consists of two MSNs which receive simulated excitatory synaptic inputs from the extracellularly recorded motor cortex units. The MSNs are reciprocally connected to each other through inhibitory synapses. The prosthetic control module monitors the spiking activity of the MSNs and determines the one with the highest firing rate as the winning neuron. The action corresponding to the winning neuron is applied by the digital base servomotor (Joint 1) of the robotic arm in one dimensional space. A global reward signal modulates the weights of the excitatory synapses between the motor cortex and medium spiny neurons through a reward-modulated spike-timing-dependent plasticity rule.

$$u' = a(bv - u) \tag{4.2}$$

where v is membrane potential, u is membrane recovery variable, v_r is resting membrane potential, v_t is threshold potential, C is membrane capacitance, a is a constant which describes time scale of u, b is a constant which describes the sensitivity of u, k is a scaling constant, η and γ are total excitatory and inhibitory synaptic conductances, respectively. E_η and E_γ represent excitatory and inhibitory synaptic reversal potentials. Arrival of a synaptic event from biological or *in silico* presynaptic neuron leads to a step-wise increase in the appropriate conductance variable; $\eta \rightarrow \eta + w_i$ for an excitatory

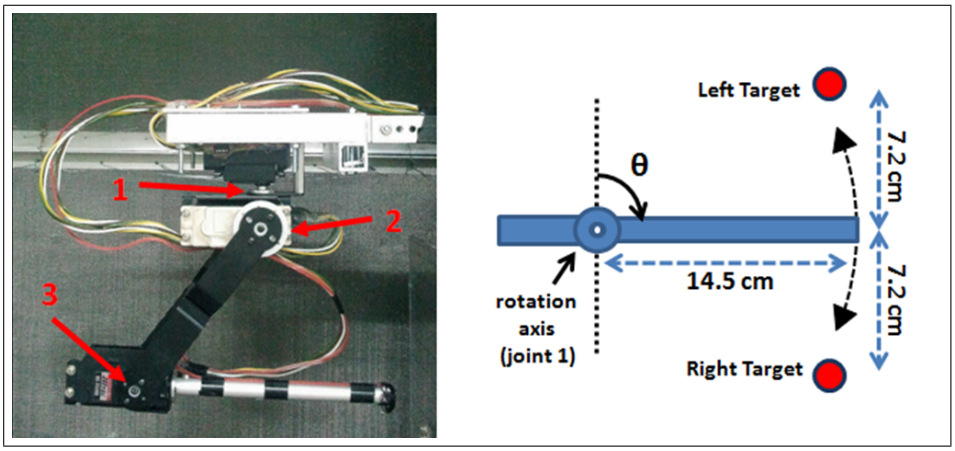

Figure 4.6 Side view and joints of the customized Lynxmotion AL5D robotic arm (left) and its representative top view in closed-loop control (right). Two opposite targets, represented with red LEDs, were located on the same plane to be reached by the robotic arm moving around its first joint (base servomotor) in one dimensional space.

event and $\gamma \rightarrow \gamma + w_i$ for an inhibitory event, where w_i is the conductance value or 'weight' of the i-th synapse of the neuron. When there is no incoming event, the total conductance values decay with time constants τ_η and τ_γ:

$$\eta' = -\eta/\tau_\eta \tag{4.3}$$

$$\gamma' = -\gamma/\tau_\gamma \tag{4.4}$$

When the membrane potential exceeds a voltage peak (v_{peak}), i.e. the neuron generates a spike, the membrane potential and membrane recovery variable are reset as follows:

$$\text{if } v \geq v_{peak}, \text{ then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \tag{4.5}$$

The neuron model parameters for the MSNs are a = 0.01, b = -20, c = -55 mV, d = 150, C = 50 pF, k = 1, v_r= -80 mV, v_t = -25 mV, v_{peak}= 40 mV [44]. The reversal potentials are E_η = 0 mV, E_γ = -110 mV and the time constants for conductance values are τ_η = 6 ms and τ_γ = 20 ms. The synaptic delays between the M1 units and the MSNs are selected from a uniform distribution between 3-5 ms and the delays between MSNs are selected from a uniform distribution ranging from 2.5-3.0 ms.

Learning in the network (or adaptation in the controller) is provided by reward-modulated spike-timing-dependent plasticity (STDP), where a global reward signal leads to long-term potentiation (LTP) or depression (LTD) in the excitatory synapses [78]. The weight (w_{ij}) of the synapse between the i-th motor cortex unit and the j-th MSN are updated every 26 ms as follows:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij} \tag{4.6}$$

$$\Delta w_{ij} = \mu w_{ij}(t) r(t) e_{ij}(t) \tag{4.7}$$

where μ is the learning rate, $e_{ij}(t)$ is the binary-stated (0 or 1) eligibility trace which is triggered when the post-synaptic node j fires after the pre-synaptic node i within a time window of 40 ms and is terminated 100 ms after being triggered [79, 80, 81, 82]. $r(t)$ is the current global reward signal evaluated as follows:

$$r(t) = (1 - \bar{R}_k) S(t) \tag{4.8}$$

where $S(t)$ is the sensory error (-1 or +1) which represents the consistency or discrepancy between the user's expected movement direction and the actual robotic action. The sensory error is extracted from the movements of the robotic actuator every 26 ms and determines the sign of the global reward signal $r(t)$. If the tip of the robot moves towards the currently selected target, the value of $S(t)$ is 1, otherwise -1. $\bar{R}_k$ is the positive reward estimate (successful target reach estimate) for the k-th target and is calculated at the end of each trial as a running mean [83]:

$$\bar{R}_k(n_k) = (1 - \frac{1}{m}) \bar{R}_k(n_k - 1) + \frac{1}{m} R_T \tag{4.9}$$

where n_k is the trial number for the corresponding target, R_T is the binary reward variable which indicates if the trial is ended with successful target reach or not (1 or 0). m is the width of the averaging window.

After updating all synaptic weights using Eq. 4.6, 'homeostatic synaptic plasticity' rule [84, 85, 86] is utilized to stabilize the excitability of the MSNs; the excitatory

synaptic weights are normalized so that the sum of all weights of excitatory synapses to the j-th MSN is kept at a constant value W:

$$w_{ij}(t+1) = \frac{w_{ij}(t+1)}{\Sigma_i w_{ij}(t+1)} W \tag{4.10}$$

In addition, the weight of the excitatory synapses is limited by a maximum value (w_{max}) in order to avoid excessive increase in a synaptic weight. The value of w_{max} is determined as follows:

$$w_{max} = \alpha(W/N_j) \tag{4.11}$$

where N_j is the total number of excitatory synapses to the j-th MSN. α is a scaling constant which is greater than 1 and determines the amount of the difference between w_{max} and the average of the weights of the excitatory synapses to the j-th MSN.

5. PROOF-OF-CONCEPT VIA REAL-TIME CLOSED-LOOP SIMULATIONS

To validate the practicality of the BMDE and study the performance of the B-BMI, we performed real-time closed-loop simulations which involved a behavioral paradigm and an external, hardware-based neural signal synthesizer. The neural signal synthesizer and the behavioral paradigm were designed to realize a full system test in which all the software and hardware modules of the BMDE are utilized. The closed-loop simulation paradigm also demonstrates an example of how to develop a bioinspired BMI controller using the BMDE.

5.1 Real-time Closed-loop Simulation Platform

In the closed-loop simulation paradigm, the model striatal neurons of the BMI controller are simulated by the SNN simulation tasks of the BMDE and extracellular recordings from primary motor cortex (M1) neurons are simulated by the neural signal synthesizer. The synthesizer, implemented using a microcontroller (Microchip PIC18F4520), provides simulated neural signals to the analog inputs of the DAQ hardware of the BMDE through its output pins (Figure 5.1). Each output pin of the synthesizer is associated with a synthetic M1 neuron and when a synthetic neuron generates a spike, the corresponding pin of the microcontroller produces an inverted 100-μs-duration pulse.

As in previous studies in which closed-loop simulations were utilized for development of reinforcement learning-based neuroprosthetic control algorithms [25, 87, 88], the synthetic neurons were created to reproduce the directional tuning properties of real M1 neurons. In the synthesizer, there are three cortical neuronal ensembles, each of which consists of six M1 neurons (Figure 5.2). The neurons of the first ensemble are tuned to the 'left' and the ones belonging to the second ensemble are tuned to the

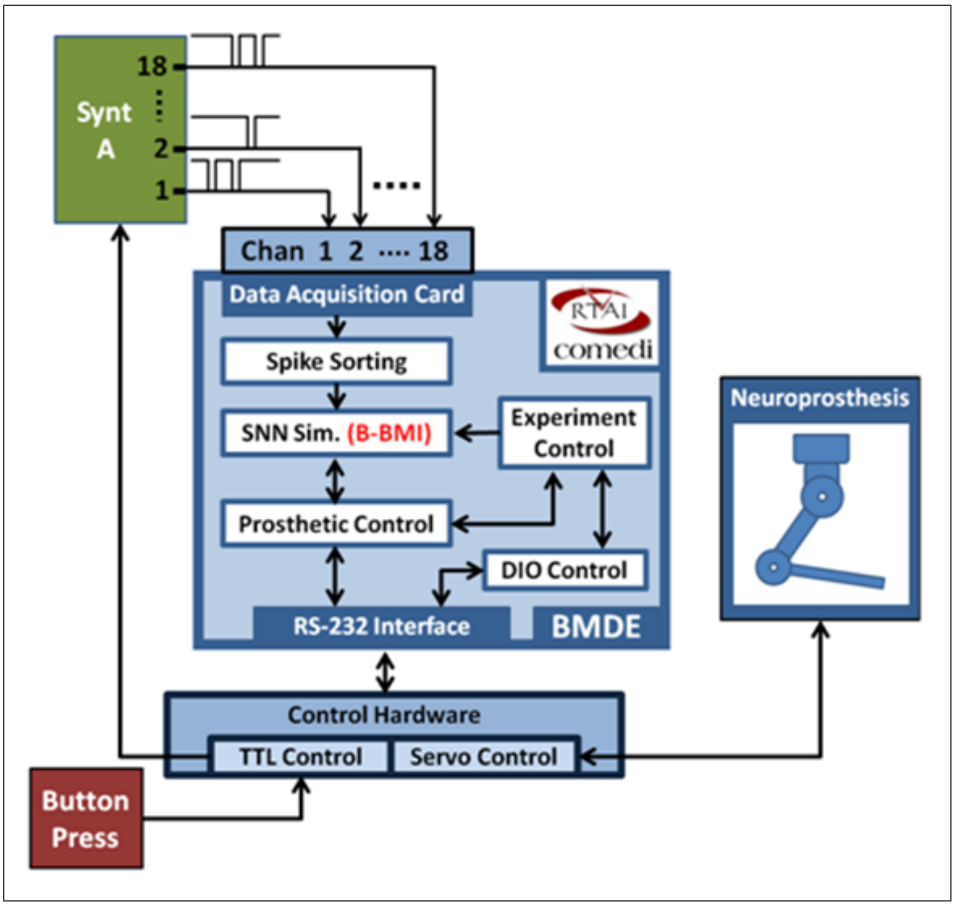

Figure 5.1 Closed-loop simulation platform for the B-BMI. The signal synthesizer (Synt-A) generates Poisson-distributed (inverted) pulse signals to simulate extracellular recordings from 18 primary motor cortex (M1) neurons and provides inputs for 18 analog channels of the DAQ device. SNN simulation task simulates the striatal MSNs of the B-BMI, which receive synapses from M1 neurons of Synt-A. A button press provides external input for the TTL control module to initiate a trial. Synt-A receives inputs from TTL control module to adjust the firing rates of the M1 neurons according to the selected target or the out-of-trial (baseline) status.

'right' direction. Additionally, the neurons of the remaining ensemble are tuned to 'no direction' as uncorrelated neurons have been observed in *in vivo* neuroprosthetic control experiments [25, 89, 90]. The M1 neurons of the signal synthesizer are connected to the *in silico* striatal MSNs through simulated synapses as explained in Figure 3.5; using the spike sorting task of the BMDE, the spikes extracted from the recordings are delivered to the *in silico* MSNs as synaptic events. As shown in Figure 5.2, two thirds of the synthetic neurons are connected to only one MSN and remaining one third are connected to both MSNs. Note that, in this configuration, some neurons with directional tuning are connected to only one MSN which represents the action towards the opposite direction (e.g. neuron 2 in Figure 5.2 is tuned to left but connected only

to right action MSN.). Thus, the M1 neurons, regardless of directional tuning, are partially connected to the MSNs through simulated excitatory synapses.

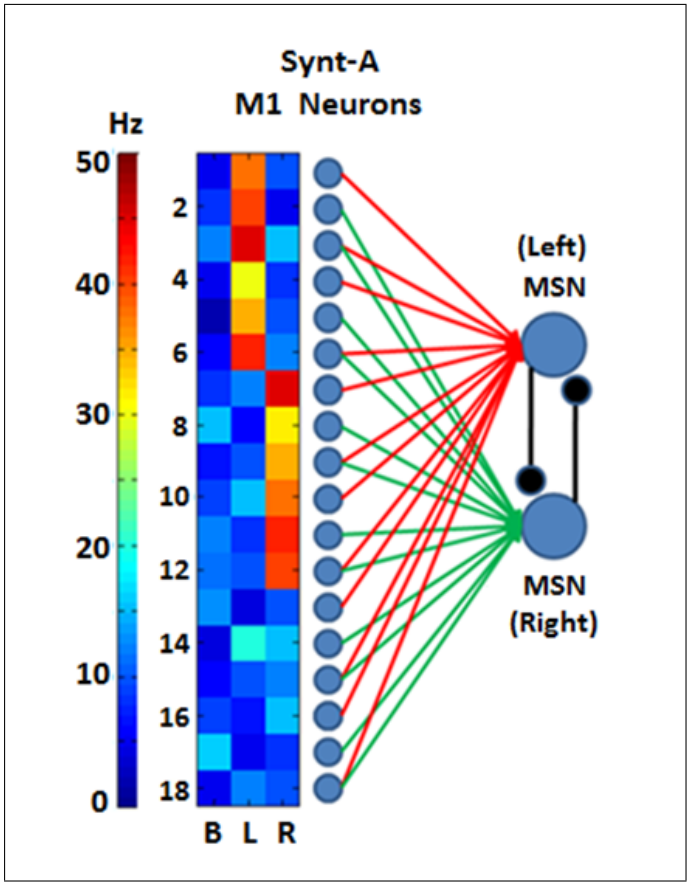

Figure 5.2 Tuning map of the M1 neurons of Synt-A (B: baseline, L: left, R: right direction) and neural network architecture for the neuroprosthetic control algorithm in closed-loop simulations. Red/green lines indicate synaptic connections between M1 neurons and left/right action MSNs.

In order to simulate the directional tuning properties of the motor cortical neurons, we programmed the synthesizer to produce Poisson-distributed spikes according to the tuning map shown in Figure 5.2. To provide the random numbers for the Poisson spike generation process, we utilized a 32-bit linear congruential pseudo-random generator [91]:

$$x(s+1) = 1664525x(s) + 1013904223 \tag{5.1}$$

where x is the generated 32-bit number, s is the number of the iteration and $x(0)$=0. The Poisson spike generation process is run every 2 ms for each synthetic neuron. The plot in the right panel in Figure 5.3 is a snapshot from the spike sorting GUI of the

BMDE while it was plotting the (digital band-pass filtered) waveforms of the neural spikes acquired from a channel of the signal synthesizer. Throughout the simulations, the synthesizer continuously generates spikes according to the baseline firing rate estimates of the neurons. When a trial starts, it begins to apply the directional tuning properties of the neurons according to the selected target.

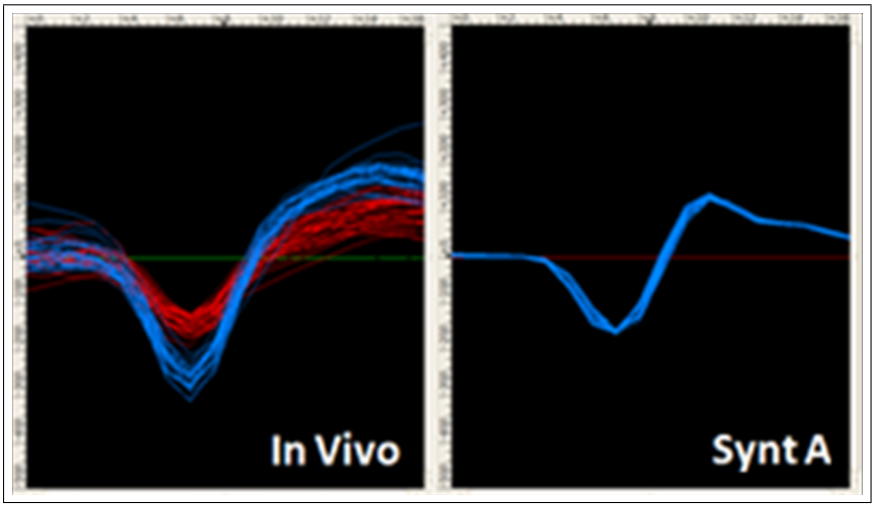

Figure 5.3 The snapshots from the spike sorting GUI of the BMDE showing the isolated spike waveforms recorded from an awake rat which was chronically implanted with a microwire array in the forelimb area of the M1 (left) and spike waveforms acquired from the neural signal synthesizer (right).

5.2 The Behavioral Paradigm for Closed-loop Simulations

According to behavioral paradigm (Figure 5.4), each trial starts by an external digital input (i.e. button press) which is provided by the experimenter at arbitrary times. When the button is pressed for 26 ms, the DIO control task of the BMDE (Figure 5.1) senses the external input through the TTL control module of the control hardware and sends a 'trial start request' message to the experiment control task, the management center of the behavioral experiments. After receiving the trial start request, the experiment control task selects a target (left or right) randomly and initiates a trial by delivering a message to the SNN simulation, the DIO control and the prosthetic control tasks. The 'trial start' message also includes the information regarding the selected target side (i.e. left or right). Thence, 1) the SNN simulation task, simulating the striatal MSNs, determines the value of the positive reward estimate

($\bar{R}_k$) corresponding to the selected target to apply Eq. 4.6 throughout the trial, 2) the prosthetic control task sets the Cartesian coordinates of the selected target to sense whether the target is acquired during a trial, 3) the DIO control task commands the TTL control module to turn the target LED on (Figure 4.6) and provides the input for the neural signal synthesizer. According to the input from the TTL control module, the neural signal synthesizer starts to generate the neural activity pattern related to the selected target (Figure 5.2).

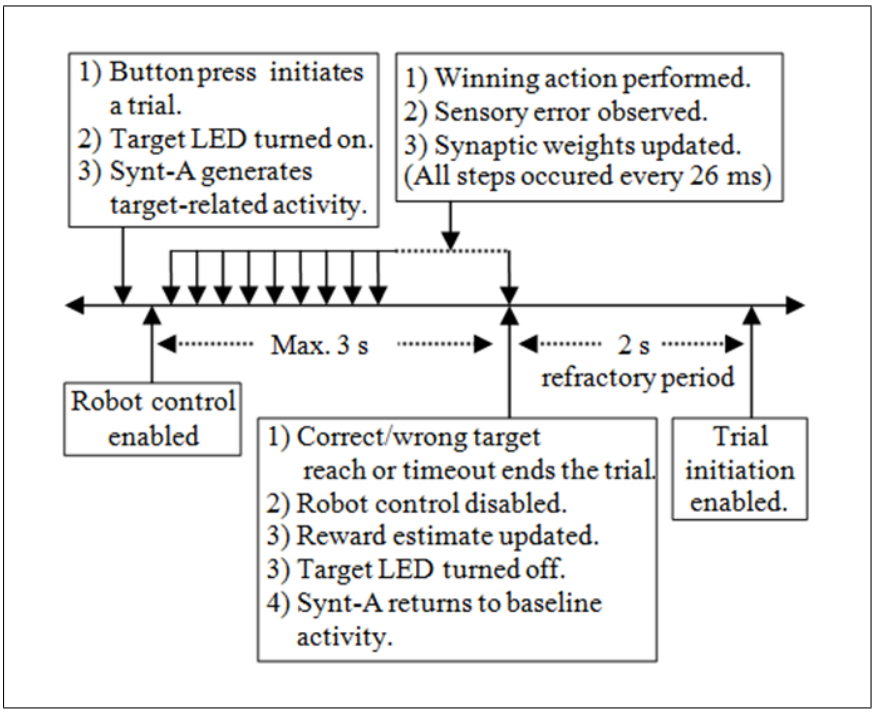

Figure 5.4 Behavioral paradigm for the closed-loop simulations.

Forty milliseconds after a trial initiates, the prosthetic control task starts to periodically handle the spike events received from the SNN simulation tasks. Every 26 ms, it calculates the spike counts of the MSNs with a sliding 104-ms time window and selects the neuron with the highest spike count as the winning neuron, whereupon it applies the action corresponding to the winning neuron by delivering a command to the servo control module of the control hardware. This command rotates the (digital) base servomotor of the robotic arm -1/+1 degrees for the winning left/right actions or keeps it stationary in case of equality among the spike count of the MSNs. Six milliseconds after delivering the pulse width command, the prosthetic control task receives the angle values of the joints through the servo control module and determines

the Cartesian position of the tip of the arm by forward kinematics. If the tip of the robot moves towards the selected target in the last movement step (26 ms), the prosthetic control task sends a positive sensory error ($S(t)$=1) message to the SNN simulation task. If not, it delivers a negative sensory error (-1). Based on the sensory error messages received from the prosthetic control task, the SNN simulation tasks periodically update the weights of the plastic synapses by applying Eq. 4.6.

Based on the Cartesian position of the tip of the robotic arm, the prosthetic control task periodically checks if the correct or wrong target is reached within the maximum trial duration (i.e. 3 s). If the correct (selected) target is reached, the prosthetic control task sends a 'reward request' message to the experiment control task. If the wrong (opposite) target is reached by the robotic arm, a 'punishment request' message is delivered to the experiment control task. By receiving such a request message, the experiment control task ends a trial by sending a message to the SNN simulation, DIO control and prosthetic control tasks. The message to the SNN simulation tasks includes the information related to correct or wrong target reach so that it can update the positive reward estimate for the selected target, $\bar{R}_k$, by applying Eq. 4.9. The message to the DIO control task cancels the inputs to the target LED and the neural signal synthesizer. Thus, the synthesizer starts to generate the baseline spiking activity for the simulated cortical neurons. Finally, by receiving the 'trial end message', the prosthetic control task directs the robotic arm back to its default position in the middle of the targets to prepare it to be used in the next trial.

In addition to the trial end messages, a trial is also terminated by the experiment control task when it is not completed by a correct or wrong target reach event within the maximum trial duration. When there appears a trial timeout, the experiment control task sends the 'trial end message' to the downstream modules as in wrong target reach case. Thus, the positive reward estimate for the selected target, $\bar{R}_k$, is decreased, DIO control task cancels the input to the neural signal synthesizer and the prosthetic control task directs the robotic arm back to the default position. At the end of each trial, a refractory period of 2 s is applied to allow the robotic arm to reach its default position and let the data writing processes to create new data folders for the

recordings related to the next trial.

Throughout the closed-loop simulations, i.e., during the trials and inter-trial periods, the spike sorting task is always enabled to extract the spike events from the acquired data using the methods explained in Chapter 3.2.1 and schedule synaptic events for the post-synaptic *in silico* neurons. Additionally, throughout the closed-loop simulations, the SNN simulation tasks evaluate the dynamics of the *in silico* cells and deliver the generated spike events to the prosthetic control task with a transmission delay of 3 ms (Figure 3.5). The prosthetic control task sorts the incoming events in the time domain and processes them only when the robotic control is enabled after a trial initiation.

For the simulation of the SNN, Parker-Sochacki (PS) method [46] is applied with the techniques presented by Stewart & Bair [47] so that full-double precision accuracy is achieved in the numerical integrations. Whenever the SNN simulation tasks are triggered by timer interrupts, they perform numerical integrations for the differential equations describing the synaptic interactions and neuronal dynamics in the system (Eq. 4.1-4.4). For the integrations, the global step size is set to 250 μs. Prior to realizing integration for a global time step, the incoming synaptic events are sorted in the time domain. As PS method allows, the global integration step size is split into local substeps separated by the incoming synaptic events (if there is any) and integration for a global step size is realized through a single 250-μs-step or multiple substeps, accordingly. Whenever a spike is generated by a neuron, an event is scheduled for the postsynaptic neuron and the prosthetic control task (Figure 3.5) with nanosecond precision, which is the precision of the system time provided by RTAI.

5.3 Learning Performance of the Bioinspired BMI Controller

Prior to starting the experiment to test the learning performance of the corticostriatal circuit-inspired BMI controller (B-BMI), we determined the total weight of the excitatory synapses to the MSNs, W in Eq. 5.1, and the weight of the lateral

inhibitory synapses between the MSNs to provide a winner-take-all functionality in the network. While the Synt-A was generating baseline spike activity according to the tuning map shown in Figure 5.2, we empirically set the value of W to 110 nS to provide a baseline firing rate of approximately 2-5 Hz for the MSNs. The weight of the inhibitory synapses, which were non-plastic throughout the experiment, was set to a value of 40 nS to provide strong lateral inhibition between the MSNs. The weight of the excitatory synapses to the MSNs was equal to each other at the beginning of the experiment and was continuously updated during the trials. The learning rate (μ) in Eq. 4.7 was 0.02 throughout the trials.

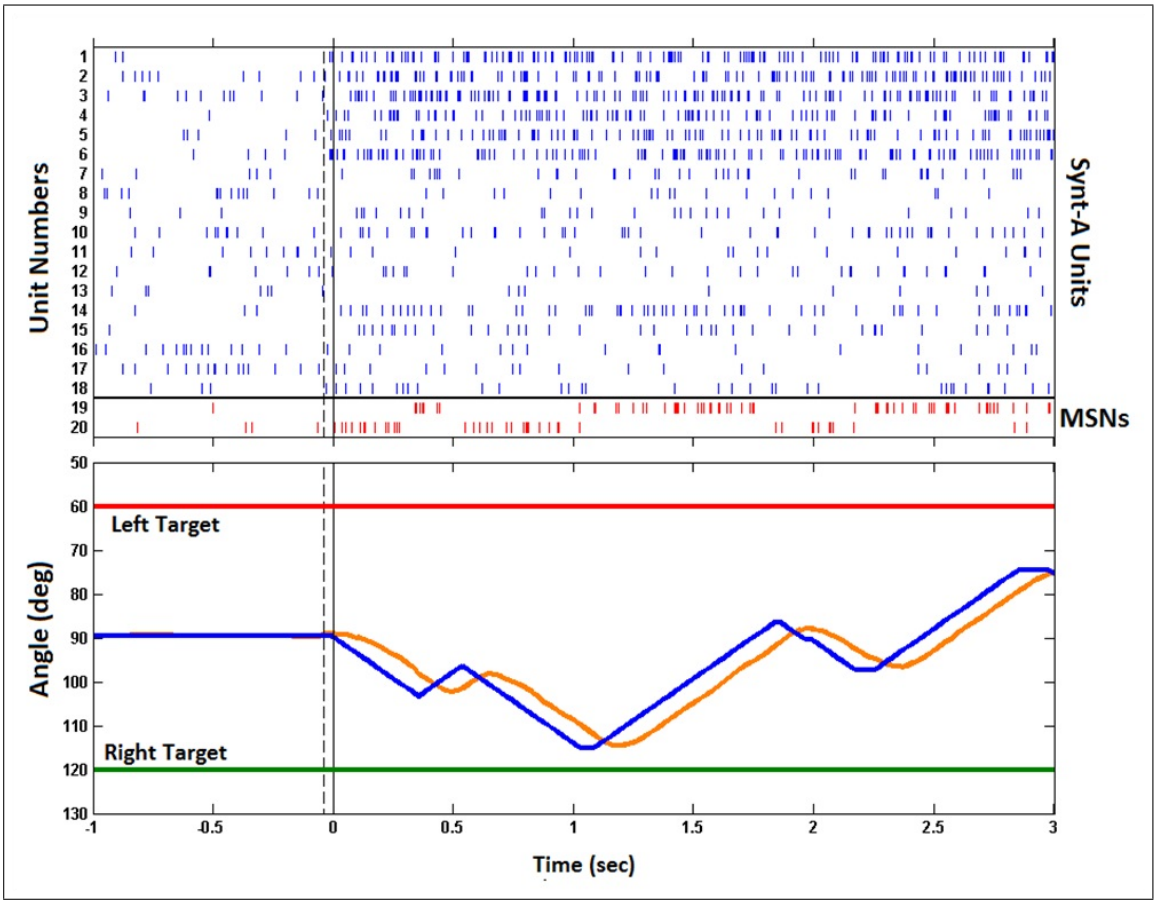

Figure 5.5 Raster of the spikes and change in joint angle modified by the outputs of the MSNs during the first trial. Upper plot shows the time of the spikes generated by Synt-A units (blue) for a selected left target and those by the MSNs (red) in response to the spike pattern generated by Synt-A. Note that the first six neurons of the Synt-A were tuned to left. The strong lateral inhibition between the MSNs can be realized through the activity pattern of the MSNs. Lower plot presents the trajectory of the robotic actuator. The blue trace indicates the joint angle values sent to the base servomotor of the robotic arm and the orange trace indicates its actual trajectory. Vertical dashed lines in both plots show the button press time to initiate a trial. Continuous black ones represent the time on which the robot control was enabled. Horizontal red/green lines present the position of the left/right targets in terms of the joint angle corresponding to the base servomotor of the robotic arm.

Figure 5.5 shows the raster plot of the spikes generated by the Synt-A and the MSNs during the first trial of the experiment. Prior to the trial, the Synt-A generated baseline spike activity. Upon initiation of the trial by a button press, the left target was selected by the experiment control task of the BMDE and the Synt-A started to produce the activity pattern for the left direction as it had been programmed to imitate the tuning properties of motor cortex neurons (Figure 5.2). The synaptic weights of the excitatory synapses to the MSNs were updated based on the sensory error extracted every 26 ms from the one-dimensional movements of the robotic arm. The trial was not rewarded since the robotic arm did not acquire the target within the maximum trial length of 3 seconds.

The learning performance of the B-BMI is shown in Figure 5.6. As the controller was naïve at the beginning of the experiment, all synaptic weights were equal and the positive reward (successful target reach) estimate for each target ($\bar{R}_k$) was zero. In this configuration, depending on the learning rate (0.02), perfect target reach accuracy was achieved after two unsuccessful trials. The positive reward estimate for each target climbed to '1' and the synaptic weights of the MSNs converged at around trial 40.

In order to test the generalization performance of the B-BMI, at trial 50, we reversed the directional tuning map of the motor cortex units; i.e., the tuning property of the units for left direction was switched to be generated for the right direction, and vice versa. In addition, the firing rate estimate of the units for the baseline activity remained the same. In this reversal learning paradigm, from the view of the MSNs, the task was to update the synaptic weights effectively to regain the perfect target reach accuracy. From Figure 5.6D, we can see a dramatic increase in the percent of the erroneously selected actions in trial 50. As the target reach accuracy of the control algorithm decreased, the reward estimate for the targets also diminished. At trial 69, target reach accuracy of the algorithm started to increase again and corresponding reward estimate value for the selected target was updated accordingly. As the reward estimate for the targets rose, the error in selected actions declined. At around trial 120, the synaptic weights converged to their final values and stayed there until the end of the experiment (Figure 5.7).

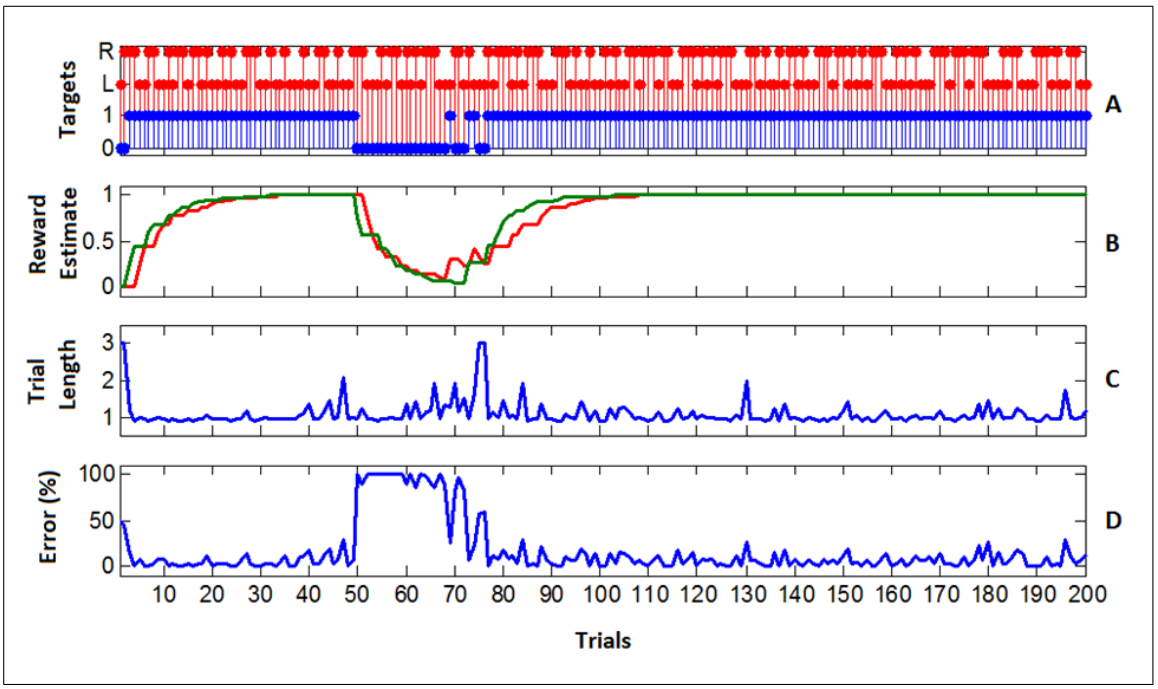

Figure 5.6 Learning performance of the B-BMI. (A) Target reach performance. The selected target for each trial is represented by red stems (L for left trial and R for right trial) and the blue stems shows if the target was acquired (1) or not (0) within maximum trial length. (B) The positive reward estimate related to each target at the end of the trials (red for left, green for right target reach-related positive reward estimate). (C) The length of the trials. The wrong or correct target reach ends the trial before its maximum allowed length (3 seconds). (D) The percent of the selected actions which does not direct robotic arm towards the selected target (in other words, the trajectory error rate). At trial 77, 100% target reach accuracy was regained for the reversed tuning map.

In order to investigate the effect of learning rate in learning speed, we reinitiated the experiment with the same parameters as in the previous experiment except for the learning rate in Eq. 4.7; we increased it from 0.02 to 0.1. As shown in Figure 5.8, with the higher learning rate, perfect target reach accuracy was achieved in the first trial since the B-BMI continuously learns within the trials. As in the previous experiment, the tuning map of the motor cortex units was reversed in order to test the generalization performance of the B-BMI. At trial 62, it regained perfect target reach accuracy as shown in Figure 5.8A.

We also monitored the weights of the excitatory synapses to the MSNs with the higher learning rate. In this setting, the convergence in synaptic weights was achieved earlier than the previous experiment (see Figures 5.7 & 5.9). At trial 50, the tuning map of the motor cortex units was reversed. Synaptic weights were re-updated to

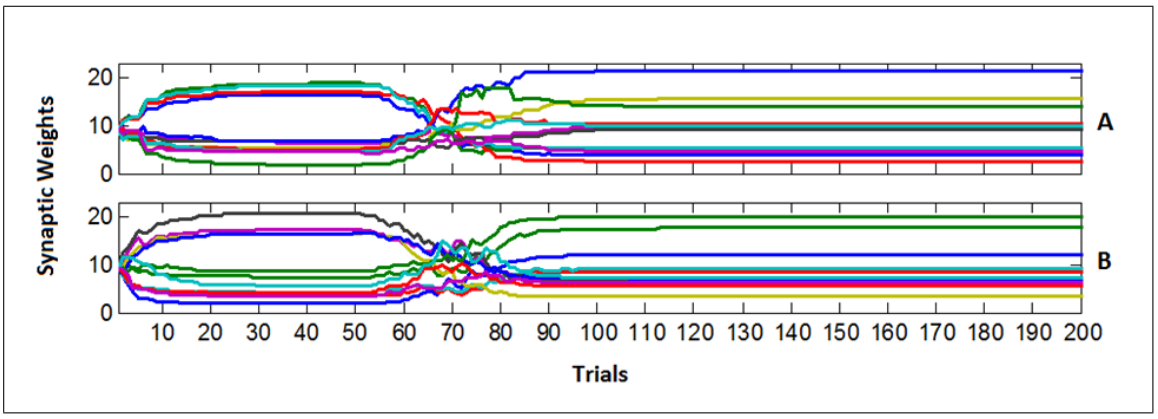

Figure 5.7 The weight of the excitatory synapses of the left (A) and right (B) action MSNs at the beginning of each trial. Convergence in synaptic weights was achieved at around trial 40 by updating the synaptic weights. At trial 50, the tuning map of the motor cortex units was reversed. Synaptic weights were re-updated to regain perfect target reach accuracy.

regain perfect target reach accuracy.

5.4 Discussion

We proposed a novel BMI controller (B-BMI) which is inspired by the synaptic interactions between the motor cortical and striatal medium spiny neurons. The controller utilizes two model medium spiny neurons (MSNs), each of which represents one of two prosthetic actions and competes with the other through strong lateral inhibition. The total weights of the excitatory synapses to each MSN and the weight of the lateral inhibitory synapses were optimized to realize a winner-take-all mechanism. In the present system, if the total excitatory weights to the MSNs are set to an excessively low value, the MSNs will not be able to reach firing rates which will be sufficient to suppress the activity of the other MSN. In contrast, if the total excitatory weights to the MSNs are set to an extremely high value without increasing the weights of the lateral inhibitory synapses, the strength of the lateral inhibition will not be sufficient to smother the activity of the other MSN. Thus, the weights of the synapses in the system should be adjusted to create a winner-take-all operation, which enables selective weight update for the eligibility-tagged synapses to one winning (active) postsynaptic neuron. The GUIs of the BMDE have been significantly useful

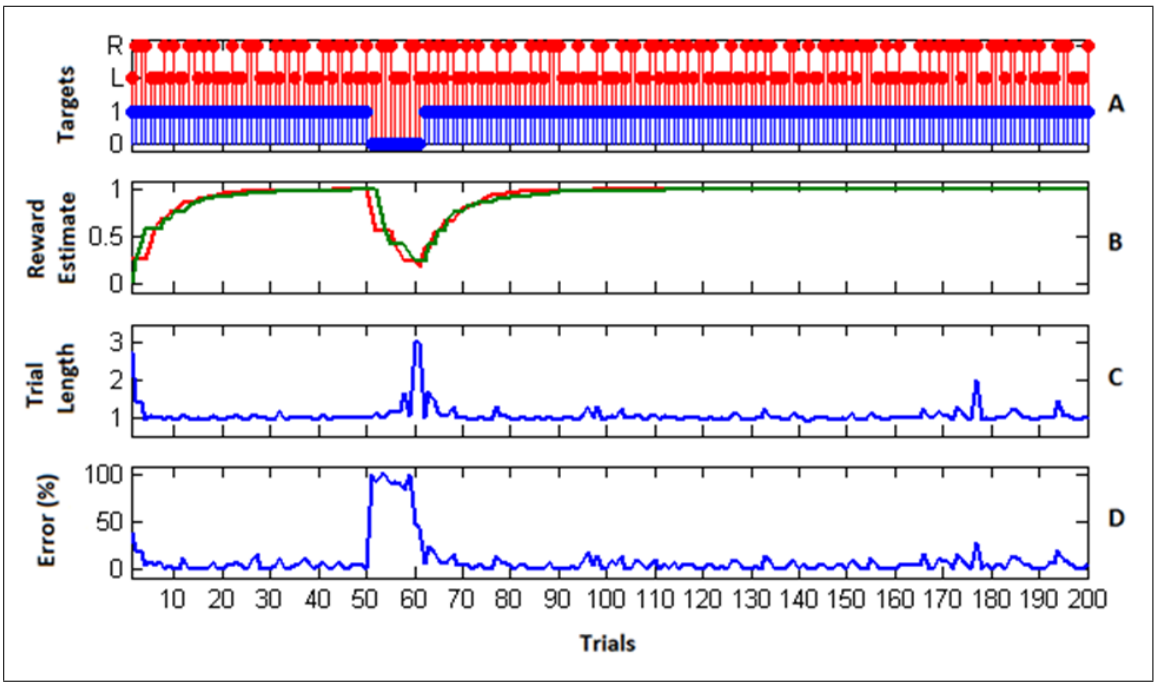

Figure 5.8 Learning performance of the B-BMI when learning rate is increased from 0.02 to 0.1. (A) Target reach performance. The selected target for each trial is represented by red stems (L for left trial and R for right trial) and the blue stems shows if the target was acquired (1) or not (0) within maximum trial length. (B) The positive reward estimate related to each target at the end of the trials (red for left, green for right target reach-related positive reward estimate). (C) The length of the trials. The wrong or correct target reach ends the trial before its maximum allowed length (3 seconds). (D) The percent of the selected actions which does not direct robotic arm towards the selected target (in other words, the trajectory error rate). At trial 62, 100% target reach accuracy was regained for the reversed tuning map.

to visualize the membrane potential dynamics of the MSNs online and optimize the synaptic weight parameters for building a winner-take-all mechanism prior to starting experiments. In this configuration, the winner-take-all operation actually sharpens the output patterns of the competing MSNs. In addition, bistable membrane potential dynamics of the MSNs enhances the hysteresis effect (Schmitt-trigger-like behavior) of the lateral inhibition in the present winner-take-all operation. The MSNs in the system are mostly inactive in the out-of-trial status (in down-state) and when a trial is initiated, the activity of winning neuron significantly increases by a transition to up-state.

Learning (or adaptation) in the present controller was achieved by reward-modulated spike-timing-dependent plasticity. The synapses leading to correlated pre- and postsynaptic activity were tagged using eligibility traces. A positive global reward

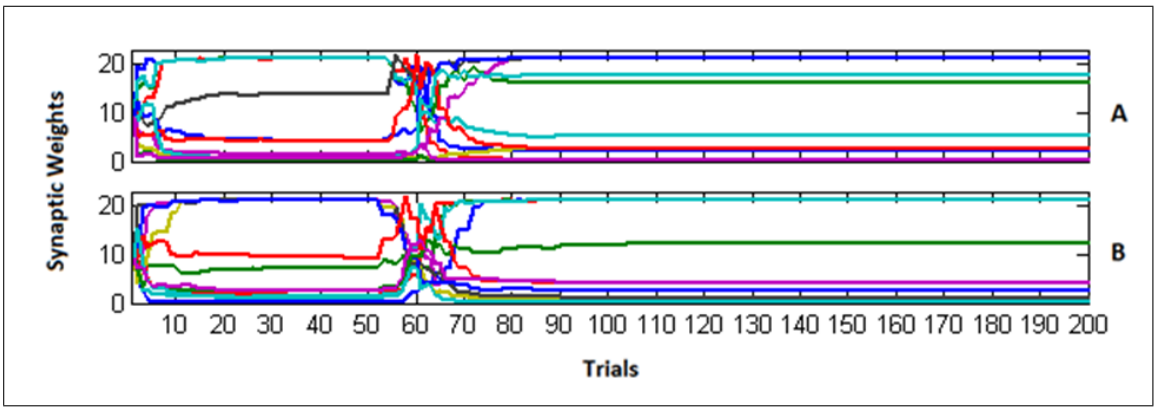

Figure 5.9 The weight of the excitatory synapses of the left (A) and right (B) action MSNs at the beginning of each trial when learning rate is increased from 0.02 to 0.1. Convergence in synaptic weights was achieved at around trial 20 by updating the synaptic weights.

signal, which may represent a phasic increase in dopamine concentration in the vicinity of MSNs, led to LTP in the eligibility-tagged synapses [67]. In contrast, a negatively signed global reward signal, which may characterize a phasic depression in the activity of the dopaminergic neurons, caused LTD in the tagged synapses [62, 92]. In the present system, the sign of the global reward signal ($r(t)$) was determined by the sensory error ($S(t)$), which was extracted from the movements of the robotic arm. The robotic movements towards the currently selected target led to an increase in reward expectancy and a positive global reward signal, and opposite-direction movements triggered a negative reward signal. Additionally, reaching behavior towards each target was treated as a different task to be learned and the system held a separate positive reward (successful target reach) estimate value ($\bar{R}_k$) for each task [78]. As the reward estimates increased by acquisition of the correct targets in consecutive trials, the magnitude of the global reward signal ($r(t)$) was degraded [93] so that the synaptic weights were automatically stabilized when perfect target reach accuracy was ensured for each target (Figures 5.6-5.9). Learning speed and convergence characteristics of the synaptic weights in the present model can be modified by changing the learning rate (μ) in Eq. 4.7 and the reward estimate averaging window size (m) in Eq. 4.9.

In the present study, the units of the neural signal synthesizer (Synt-A) simulated the spiking behavior of motor cortical (excitatory) regular spiking neurons. During the inter-trial periods, the units generated baseline activity which was charac-

terized by low frequency spike trains and, during the trials, the units with directional tuning increased their spiking activity according to selected target [94] (Figure 5.2). As it is possible to distinguish the (excitatory) regular spiking units from (inhibitory) fast spiking ones through electrophysiological recordings [94, 95], we foresee that it will be possible to provide artificial excitatory synaptic inputs to the simulated MSNs from the cortical regular spiking units during future *in vivo* neuroprosthetic control studies. In addition, throughout the closed-loop simulations, the tuning map for the units of the neural signal synthesizer was static; the tuning properties of the units did not alter by experienced rewards. Since the motor cortex neurons have the capability to adapt their activity patterns for efficient control of neuroprostheses [31, 96], we expect that cortical neuroplasticity will have a positive effect on the performance of the developed B-BMI [20].

In the B-BMI, we utilized strong lateral inhibitory synapses between the MSNs in order to create a winner-take-all type competition. In reality, however, the equilibrium potential for Cl- (considered to correspond to E_γ in Eq. 4.1) in the MSNs is higher than the resting membrane potential. In fact, it is close to that of the up-state. Thus, the GABAergic inputs through the axon collaterals between the MSNs assist the up-state [97]. The inhibitions for the MSNs are thought to mainly come from feed-forward fast spiking interneurons which are small in number and stronger in inhibitory effect [98]. These neurons are called feed-forward since they receive excitatory input from the cortical areas and deliver their axons to MSNs. The other component which significantly affects the excitability of the MSNs is the dopaminergic projections from the substantia nigra pars compacta and the ventral tegmental area. While the activation of D1 receptors on the MSNs facilitates transition from down-state to up-state, activation of D2 receptors makes them less sensitive to excitatory inputs [99]. In the B-BMI, such effects of GABAergic inputs from neighboring MSNs and fast-spiking interneurons, and dopaminergic inputs from midbrain structures are not considered. Instead, we built an adaptive BMI controller which is inspired by the dopamine-dependent plasticity in the corticostriatal synapses and utilizes the bistable (up & down states) behavior of the medium spiny neurons. Development of action selection (a function often attributed to basal ganglia, whose entry point is the striatum) mechanisms [100, 101] which are

based on biologically realistic models of corticostriatal or cortico-basal ganglia circuit and utilize them as a BMI controller is a matter of further research.

6. PERFORMANCE PROFILES OF THE BMDE

6.1 Stress Test Methods

In order to evaluate the real-time computational capacity of the BMDE for its future uses in the design of more complex BMI controllers which utilizes a higher of neurons or simulation parameters, we developed and implemented a stress test paradigm. In this paradigm, we utilized the same closed-loop simulation setup (Chapter 5.1) and the behavioral paradigm (Chapter 5.2) as the one used in studying the convergence properties of the B-BMI. While the B-BMI was implemented on the BMDE, we additionally inserted 150 MSNs into the SNN and connected an extra neural signal synthesizer (Synt-B) to the remaining 14 channels of the DAQ board (Figure 6.1). These additional neurons and the signal synthesizer had no function in neuroprosthetic control; they were added into the closed-loop simulation platform only for stressing the hardware and software resources of the BMDE while the proposed controller, the B-BMI, was learning the control of the robotic arm. In this scenario, the additional MSNs received inputs both from Synt-A and Synt-B through probabilistically connected excitatory synapses (connection probability = 0.66, the synaptic delays were selected from a uniform distribution ranging from 3-5 ms.). Moreover, these neurons were connected to each other through inhibitory synapses with a connection probability of 0.2; the transmission delays were selected from a uniform distribution between 2.5-3.0 ms. Consequently, each of the additional MSNs received approximately a total of 50 synapses from each other and from the signal synthesizers. Throughout the stress tests, each unit of the Synt-B continuously generated Poisson-distributed spikes with an estimated firing rate of 80 Hz independent from trial initiation or termination.

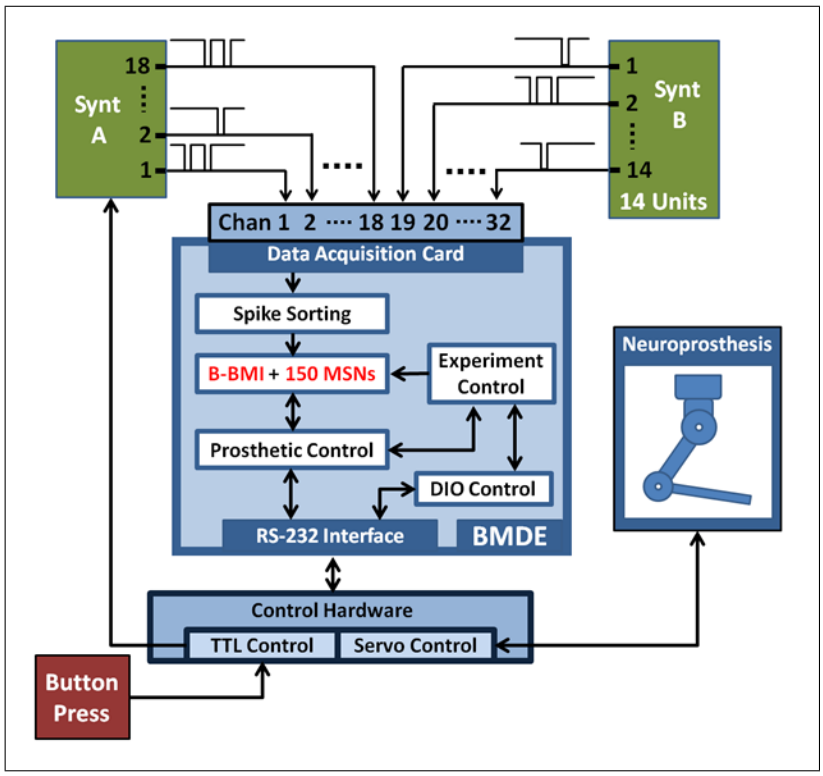

Figure 6.1 The stress test platform. For the stress test, an additional 14 channel neural signal synthesizer (Synt-B) and 150 MSNs are added into the closed-loop simulation paradigm shown in Figure 5.1. These MSNs receive inputs from Synt-A and Synt-B through probabilistically connected synapses, but do not provide any output which can affect the operation of the B-BMI.

6.2 Results

In order to evaluate the performance of the BNDE, we ran two test cases: Firstly, we ran the B-BMI using the closed-loop simulation methods explained in Chapter 5.1 ('only B-BMI paradigm'). Secondly, we ran the stress test paradigm explained in Chapter 6.1 ('B-BMI+150 MSNs'). Each test case was run for two hours while online visualization and data recording tools of the BNDE were enabled. For each test case, 500 trials were performed on the system. During the tests, the spike sorting task ran template matching algorithm for three templates for each channel. One template was generated based on the spike waveform produced by the neural signal synthesizers (Figure 5.3) and the remaining two were produced by applying a sinusoidal waveform to the input channels of the DAQ hardware. Additionally, at trial 50 of each test, the tuning map for the motor cortex units were reversed as mentioned in the Chapter 5.3.

At the end of both tests, the SNN simulation task and the prosthetic control task did not miss any spike event generated by the neural signal synthesizers (Synt-A&B) and the SNN simulation tasks; all spike events extracted by the spike sorting task were processed by the SNN simulation tasks on their scheduled time and prosthetic control task handled all spike events in the corresponding time bin (Figure 5.3). During these tests, we observed execution times of the RT tasks and deviations in their periods (i.e. jitters) in order to evaluate the real-time performance of the system. Based on measurements through 100,000 consecutive execution cycles, average jitter for the tasks triggered by the timer interrupts of the system (i.e. all tasks except for spike sorting task) was less than 1 μs, with a maximum of 20-25 μs. For the spike sorting task, which was triggered by the interrupts of the DAQ hardware, the average jitter was approximately 9 μs, with a maximum of 110-120 μs. Despite occurrence of such jitters in execution of the spike sorting task, the times of the spikes could be deterministically determined through the recordings since the preconfigured interrupt generation period (512 μs) and the number of samples per DAQ period (16 samples per channel) were known in the system.

Table 6.1 presents the average and maximum execution times of the RT tasks during both tests based on 100,000 execution cycles. From the table, we can see that increasing the number of the MSNs for the stress test also increased the execution time of the SNN simulation tasks. As expected, by addition of the Synt-B into the simulation platform for the stress test paradigm, the average execution time for the spike sorting task also increased due to operation of template matching algorithm for additional 14 DAQ channels. However, the execution time for other tasks was not significantly affected by such increases since they were run on different cores of the CPU.

During running only the B-BMI and the stress test paradigm, the BMDE recorded 1) the timestamps of the spikes generated by both the biological and *in silico* neurons, 2) the input and output events related DIO control task, 3) the pulse width commands sent to the servo control module, 4) joint angle values received from the servo control module and 5) the statistics related to the experiments. At the end of

Table 6.1
Execution times of the RT tasks during running only the B-BMI and the stress test.

	B-BMI Only		Stress Test (B-BMI + 150 MSNs)		
RT Task	Average (μs)	Maximum (μs)	Average (μs)	Maximum (μs)	Task Period (μs)
Spike Sorting	59	145	71	188	512
SNN Simulation 0	7	22	767	1614	2000
SNN Simulation 1	7	20	770	1578	2000
Prosthetic Control	5	55	6	61	2000
Experiment Control	2	31	2	28	2000
DIO Control	3	15	3	15	2000

the performance test for the B-BMI, the BMDE recorded 48 MB of data. In the case of the stress test, as the number of the spiking units was increased by an addition of a neural signal synthesizer (Synt-B) and 150 MSNs to the SNN simulator, the BMDE recorded 1.18 GB of data in two hours. During the stress tests, the average spiking activity per simulated MSN was approximately 41 Hz.

During the performance tests for running only the B-BMI and the stress test paradigm, we were able to monitor the plot of the dynamics of the MSNs and raster of the spike events generated by the neural signal synthesizers (Synt A&B). Figure 6.2 shows a snapshot from the GUI of the BMDE while it was plotting the raster of the spikes online during the 462nd trial of the stress test. Additionally, Figure 6.3 shows the snapshot of the GUI while plotting the dynamics of three manually selected MSNs during the same trial. The uppermost graph in Figure 6.3 illustrates the membrane potential dynamics of the MSN which corresponds to the left prosthetic action and the graph in the middle presents those of the MSN corresponding to the right prosthetic action. Finally, the bottom graph shows the high-frequency spiking activity of one of the 150 MSNs added into the system for the stress tests. The green vertical lines in Figures 6.2 & 6.3 mark the timepoint on which the control of the robotic actuator was enabled and the red lines indicate the timepoint on which the trial was ended by

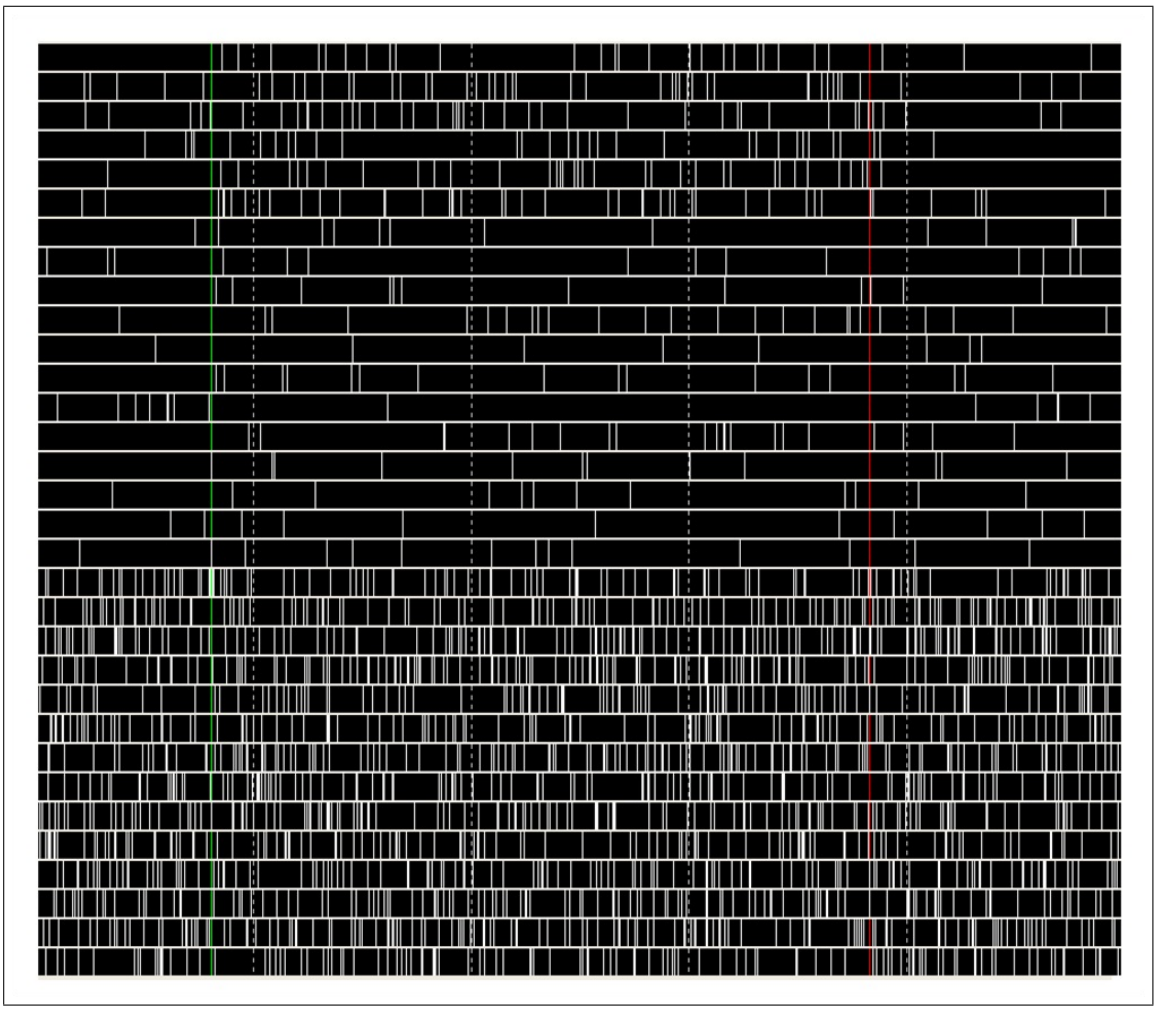

Figure 6.2 Raster plot of the spikes generated by Synt-A&B during 462nd trial of the stress test. The first six rows present the tuned activity of the neurons simulated by the Synt-A. Last 14 rows reflect the activity pattern of the high-frequency spiking neurons of the Synt-B.

acquisition of the right target. Even though the GUIs are capable of plotting 3-second history of the raster of the spikes and the neuronal dynamics, we took the snapshot of the same 1.5 second-portion of the visualized part in order to provide a higher resolution image here. The time interval between the vertical dashed lines in Figures 6.2 & 6.3 is 300 milliseconds and the voltage difference between the horizontal dashed lines in Figure 6.3 is 30 mV.

6.3 Discussion

In the present work, one of our goals was to implement a platform capable of creating simulated synaptic connections from extracellularly recorded neurons to model neurons for development of SNN-based BMI controllers. Since the software-based SNN simulations provide a flexible method for investigating the behavior of the neuronal circuits, we preferred to develop this platform around a desktop PC. In order to guarantee the strict timing constraints of the real-time SNN simulations and biological/*in silico* neuronal interactions, we used RTAI, a real-time extension for Linux operating system. Utilization of RTAI provided several benefits in establishment process of the modeling environment. First, it enabled development of real-time applications which are equipped with powerful GUIs capable of live visualization of the spiking activity of the biological and *in silico* neurons (Figures 6.2 & 6.3). Second, by the support of the COMEDI drivers, the RTAI provides deterministic response to the interrupts of a DAQ device. Thus, the neural data acquired from the brain or neural signals synthesizers (Figure 5.3) could be processed in real-time. Third, since the RTAI provides serial port drivers, it became possible to perform the control of a robotic actuator according to the outputs of a SNN which was trained using the position-related feedbacks received from the actuator. Finally, the use of the open-source software provided by RTAI and COMEDI projects significantly decreased the costs in establishment of the *in vivo* modeling environment presented here.

The other motivation for this study was to validate the real-time performance of the BMDE for its use in animal experiments. In order to test all its implemented software components, we connected an external, hardware-based neural signal synthesizer to the analog input channels of the DAQ hardware and developed the Bioinspired BMI controller (B-BMI) using the simulated cortical inputs provided by the signal synthesizer. During the development process, the B-BMI was interfaced with a robotic arm operating in real-world. Using the behavioral paradigm presented in Figure 5.4, which involved external binary inputs (e.g. button press) to initiate neuroprosthetic control trials and outputs to indicate the position of the targets (e.g. LED targets), the system learned the control of the robotic arm for a two target reaching task in one

dimensional space. In these simulations, the spike sorting task of the BMDE sorted the detected spikes into one of three synthetic units by applying the Gaussian template matching algorithm presented in Chapter 3.2.1. In addition to the real-time closed-loop simulations realized using the neural signal synthesizers, the classification performance of the Gaussian template matching-based spike sorting utility of the system was also validated by performing *in vivo* neural recordings from the rat motor cortex (Figure 5.3). Thus, we confirmed the real-time performance of the BMDE for its use in future *in vivo* neuroprosthetic control experiments.

Since we plan to utilize the BMDE in the long run for development of large scale models of motor-related circuits consisting of large number of neurons, we examined its performance by a stress test paradigm involving simulation of 150 neurons which received dense synaptic connections from each other and the spiking units of the signal synthesizers (Chapter 6.1). Throughout the stress test, the BMDE was capable of manipulation of the neuroprosthesis using the B-BMI while simulating additional 150 MSNs without missing any spike event generated by the neural signal synthesizers or the simulated MSNs.

Since the COMEDI project provides drivers for a variety of DAQ boards, the BMDE can be implemented using products of other vendors as long as the drivers for those boards support adequate sampling frequency for spike sorting. In addition, the DAQ boards should be capable of being programmed to deliver on-board buffered data to the system memory periodically and generate an interrupt at the end of data delivery process so that the spike sorting task can be triggered in response to the interrupts. Moreover, the DAQ device which is to be utilized should be capable of buffering an adequate amount of samples into its on-board memory and deliver them to the system memory at an appropriate frequency; if the data transmission period, and the resultant interrupt generation period, for the DAQ device is too low, then the GUIs of the system might be unusable due to consumption of the system resources for processing these excessively frequent high priority hardware interrupts. We determined an appropriate period (512 μs corresponding to 16 scans) for the DAQ interrupt generation in the BMDE so that all the components of the system worked smoothly.

In the present implementation of the BMDE, we used a quad-core PC and assigned the SNN simulation process to two cores of the CPU. As the software architecture of the BMDE and RTAI allows the SNN to be simulated by multiple RT tasks assigned to different cores of the CPU, the number of neurons which can be simulated by the system can be improved by utilization of CPUs consisting of a higher number of cores. Additionally, in the present study, we performed SNN simulations with double-precision integration accuracy using the Parker-Sochacki method. By sacrificing the accuracy in numerical integrations, higher number of neurons can be simulated in the system using well-known integration methods such as Euler and Runge-Kutta methods [47]. As a further improvement, since the RTAI and COMEDI libraries enable utilization of multiple DAQ devices on a single PC, additional DAQ devices can be inserted into the system so that the number of units isolated from the neural recordings can be increased to provide a higher number of synaptic connections to the model neurons from the real neurons.

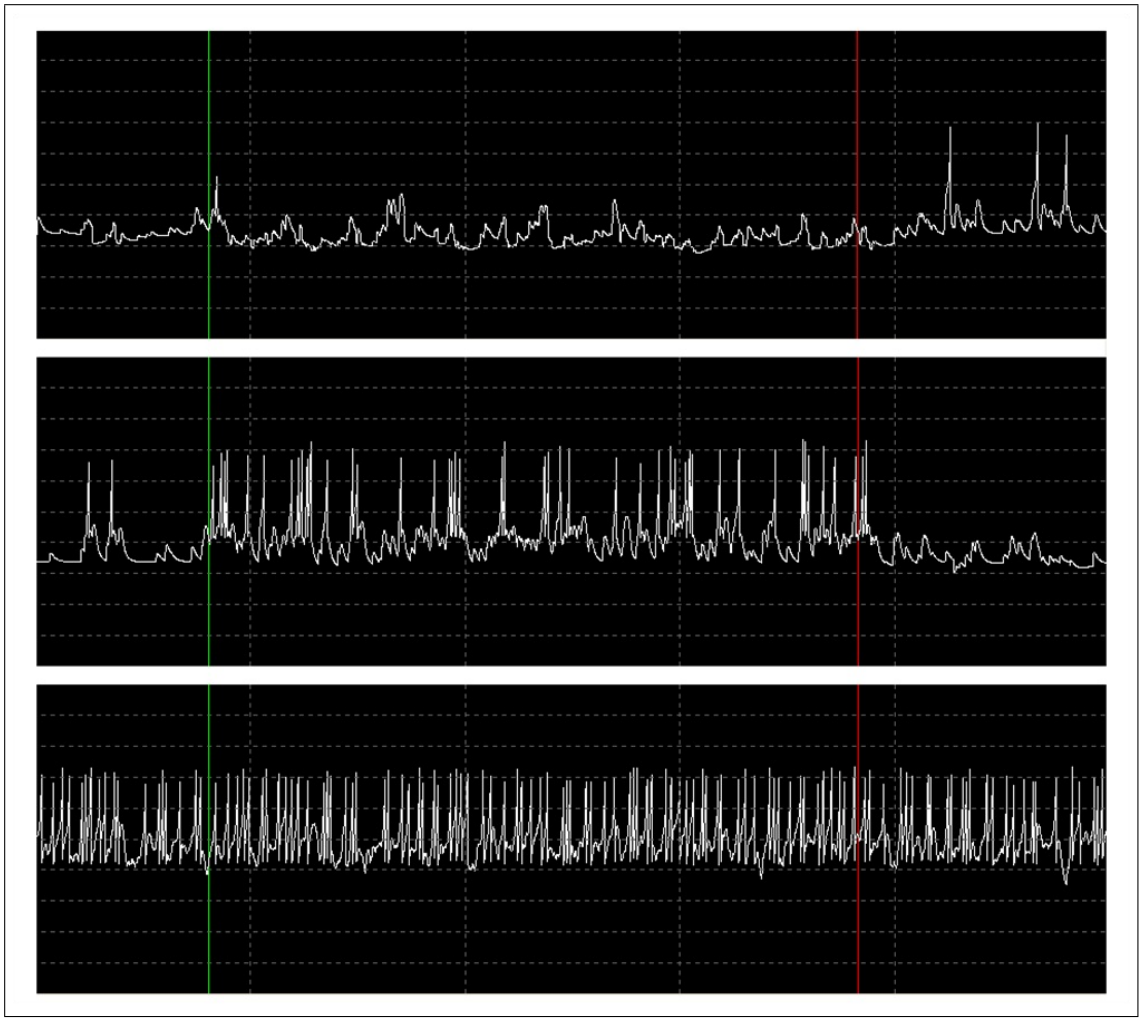

Figure 6.3 Dynamics of the MSNs during 462nd trial of the stress test in response to the spike events presented in Figure 6.2. When the trial was started and the robot control was enabled (vertical green line), the MSN corresponding to right action was activated by the increasing activity of the simulated motor cortex units (graph in the middle) and suppressed the activity of the left action MSN (uppermost graph) through lateral inhibition. At the end of the trial by successful target reach (vertical red line), the MSNs returned to their baseline activities with the decrease in the activities of the presynaptic motor cortex units. The bottom graph represents the neuronal dynamics of one the MSNs which were added into SNN for the stress test paradigm.

7. CONCLUSIONS

7.1 Novel Contributions

The work in this dissertation brings a novel, bioinspired motor neuroprosthetic design approach, which is based on building simulated synaptic connections between real neurons and model neurons. In order to fulfill the challenging real-time requirements of this approach, we first implemented the Bioinspired Model Development Environment (BMDE). Using this BMDE, we developed a novel BMI controller (the B-BMI).

The BMDE (presented in Chapter 3) provides a low-cost, all-in-one and extendible solution for the development of new, bioinspired neuroprosthetic systems capable of realizing the control of a robotic arm through building hybrid biological/*in silico* neural networks. In the hybrid neural network, extracellularly recorded real neurons are connected to the model neurons via simulated (virtual) synaptic connections using extracellular recording techniques. While time-critical biological-*in silico* neuronal interactions and neuroprosthetic control are realized in the background, the BMDE allows the neuroprosthetic designer to visualize the dynamics of the hybrid neural network online and manage the behavioral experiments through graphical user interfaces (GUIs). In this way, it enables the designer to optimize the parameters of the hybrid neural network during neuroprosthetic control experiments. It also supports spike data recording related to real and simulated neurons for analyzing the dynamics of the BMI controller. Using the features of the BMDE, we developed the bioinspired BMI controller (the B-BMI), based on utilization of two model striatal medium spiny neurons.

The BMDE was shown to be capable of simulating 150 Izhikevich neurons (each receiving approximately 50 synapses) in real-time with double-precision integration accuracy while running the B-BMI and manipulating the robotic arm. This perfor-

mance profile indicates that the BMDE can support more sophisticated SNN-based BMI control algorithms. Since the BMDE was developed on RTAI, which is a real-time extension for Linux operating system, it can be modified easily to run on a wide variety of PC hardware platforms.

The control architecture of the neuroprosthetic system (the B-BMI), described in this dissertation (Chapter 4), is fundamentally distinct from those of conventional systems. Most of the conventional neuroprosthetic systems are designed from the perspective of input-output mathematical modeling; the main design motivation is generally to find a mathematical model or transform which optimally maps motor cortical activity into user's intended prosthetic actions. In these systems, a 'spike binning' preprocess is also performed to provide cortical firing rate inputs to the input-output model used and this preprocessing leads to loss in the information encoded by timing of the spikes. In contrast, in the present system, the control of the neuroprosthesis is realized in a more biologically plausible manner. Cortical spike events are directly forwarded to the model striatal neurons through simulated synapses without any preprocessing. There is no spike binning therefore the neural information provided by spike timing [102, 103, 104] at the input stage of the BMI controller is lossless. Adaptation or learning in the present system is realized by simulating a possible mechanism of corticostriatal plasticity; a reward signal, modeling phasic changes in dopaminergic signaling in the striatum, is used to update the weights of the eligibility-tagged synapses between the motor cortical and model striatal neurons. In conclusion, the proposed control algorithm brings a novel and more biologically plausible approach for neuroprosthetic design.

7.2 Implications

The work in this dissertation aims to shift the design approach for BMI controllers from standard input-output modeling to building hybrid neural networks, where real neurons are coupled to *in silico* (model) neurons through simulated synapses to provide more biologically plausible neuroprosthetic control. The present design ap-

proach can enable us to develop new BMI controllers which are inspired by the circuits of the brain. Since these bioinspired control algorithms, built on model neurons, can directly receive excitatory/inhibitory or other type of simulated synaptic inputs from the brain circuits, these systems have the potential to interact with the nervous tissue better than the conventional neuroprosthetic systems. In conventional systems, for example, the type of the neurons from which extracellular recordings are performed is not an important design concern. However, in the B-BMI, the extracellularly recorded motor cortical regular spiking (excitatory) neurons can be selectively and partially connected to the model MSNs through simulated excitatory synapses as in natural corticostriatal circuit. Moreover, the synaptic weights between these neurons can be updated by modeling a possible mechanism of STDP in the corticostriatal circuit.

Investigating the simulated synaptic interactions between the real motor cortex neurons and model striatal neurons during a neuroprosthetic control task can provide new insights into the information processing principles in the neuronal circuits involved in motor control and learning.

The bioinspired BMI, presented in this dissertation, is capable of continuously adapting its parameters in order to guarantee perfect target reach accuracy in one dimensional space for a two target center-out reaching task. Its design is based on co-adaptation of two intelligent systems: 1) the B-BMI and 2) the brain. The B-BMI is capable of changing the weights of its simulated synapses to provide adaptation to changing activity patterns of the motor cortex neurons. In this paradigm, the motor cortex neurons can also change their activity patterns to improve the neuroprosthetic control performance. In other words, two intelligent agents share the same goal and aims to adapt themselves to the changing conditions of each other. Learning rate in Eq. 4.7 determines the bias of the adaptation between these systems. When it is decreased, the contribution of the brain, by neuroplasticity, will be more pronounced in the neuroprosthetic control. In contrast, increasing the learning rate will boost the adaptability of the B-BMI.

In a clinical setting, the B-BMI does not require a training data set due its adap-

tation feature. On the other hand, most conventional systems require a training data set since, in general, they use supervised learning methods. Using only one external reward signal to the B-BMI can be sufficient to accurately control a neuroprosthesis. If such a reward signal can be identified from the brain tissue of the patient, then the B-BMI can be a fully autonomous system.

The Bioinspired Model Development Environment (BMDE) is capable of live visualization of neuronal dynamics of *in silico* neurons, and spike patterns and waveforms of real neurons while time critical biological-*in silico* neuronal interactions and control of the robotic arm is realized in the background. The real-time tasks of the BMDE (Figure 3.2) provide a framework for development of neuroprosthetic controllers. In this context, their content can be modified according to the requirements of the BMI controller and the behavioral paradigm.

The number of simultaneous SNN simulation tasks can be increased using a system equipped with a CPU with a higher number of cores. We developed the GUIs required for development of the B-BMI; new GUIs can easily be modified for different neuroprosthetic development applications.

7.3 Future Directions

We presented the BMDE as a practical and powerful platform for developing biologically inspired neuroprosthetic systems. Additionally, we proposed a novel BMI controller which was designed on the BMDE using real-time closed-loop simulations. Performance profiles of these simulations, involving a behavioral paradigm and an external neural signal synthesizer, not only show that the BMDE is capable of creating simulated synaptic connections from real neurons to *in silico* neurons during behavioral experiments but also present an important proof-of-concept for biologically plausible neuroprosthetic control. In this context, the BMDE and the proposed BMI controller (the B-BMI) can be utilized in future *in vivo* neuroprosthetic control experiments and the interactions between real motor cortical and model medium spiny neurons can

be investigated. The simulated synaptic interactions between cortical and *in silico* medium spiny neurons during neuroprosthetic control can provide new and significant knowledge about spike timing properties of the neurons in the primary motor cortex [102, 104, 105, 106].

Adaptation in the proposed controller is realized in a biologically plausible manner using a reward-modulated spike-timing-dependent plasticity rule; a reward signal, modeling phasic changes in dopaminergic signaling in the striatum, is used to update the weights of the synapses leading to correlated pre- and postsynaptic activity. If such a reward signal can be identified from the dopaminergic neuronal activity or dopamine concentration in the vicinity of corticostriatal synapses [107], a fully self-learning, autonomous BMI controller, which does not require any external training signal, can be implemented. Further research is required to investigate the dopaminergic activity during neuroprosthetic control and learning.

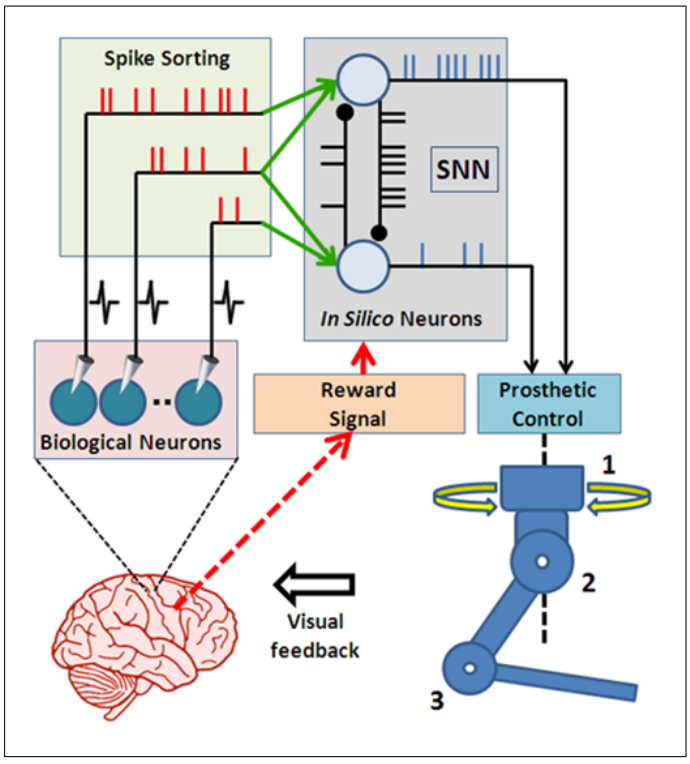

Figure 7.1 Reward signal, extracted directly from the brain, for autonomous neuroprosthetic adaptation.

In the proposed B-BMI, neural decoding is realized by building a winner-takes-all (WTA) mechanism, by connecting the striatal medium spiny neurons with strong lateral inhibitory synapses. The B-BMI, based on WTA, essentially performs action selection, a function also often ascribed to basal ganglia. Developing a biologically realistic model of the striatum or basal ganglia was out of the scope of the work in this dissertation; we proposed the B-BMI as a neuroprosthetic controller which is inspired by the motor corticostriatal circuit. Development of a realistic model of the corticostriatal or cortico-basal ganglia circuit for action selection and utilization of this model in controlling a neuroprosthesis requires further research. As it supports simulation of 150 MSNs (with double precision integration accuracy), we believe that more biologically plausible BMI controllers can be developed using the BMDE. In addition, for the development of more realistic models, the optimal number of extracellularly recorded neurons must be determined. The number of neural recording channels in the BMDE can be increased using COMEDI driver libraries, which support insertion of additional data acquisition boards into the system.

The number of simultaneous SNN simulation tasks can be increased using a system equipped with a CPU with a higher number of cores. We developed the GUIs required for development of the B-BMI; new GUIs can easily be modified for different neuroprosthetic development applications.

We developed the BMDE around RTAI. Since RTAI is a real-time extension for Linux operating system, the BMDE can be implemented on various platforms equipped with more powerful CPUs. In fact, by utilization of RTnet (Hard Real-Time Networking for Real-Time Linux, www.rtnet.org), multiple PCs could be connected to each other through hard real-time network protocols to simulate a large-scale SNN in real-time to control a neuroprosthesis using biologically realistic models of the motor-related neuronal circuits.

Since the model neurons are event-driven computing units, their utilization in neuroprosthetic design can enable development of bidirectional BMIs. RTAI and COMEDI libraries utilized in developing the BMDE also support hard real-time con-

trol of the digital outputs of data acquisition devices. Therefore, using the BMDE, it would be possible to control some optical [108] or electrical stimulation [109] devices through the digital output channels of the data acquisition device. The spike outputs of the SNN simulation task of the BMDE could then be used to stimulate the brain tissue for developing bidirectional BMIs as shown in Figure 7.2.

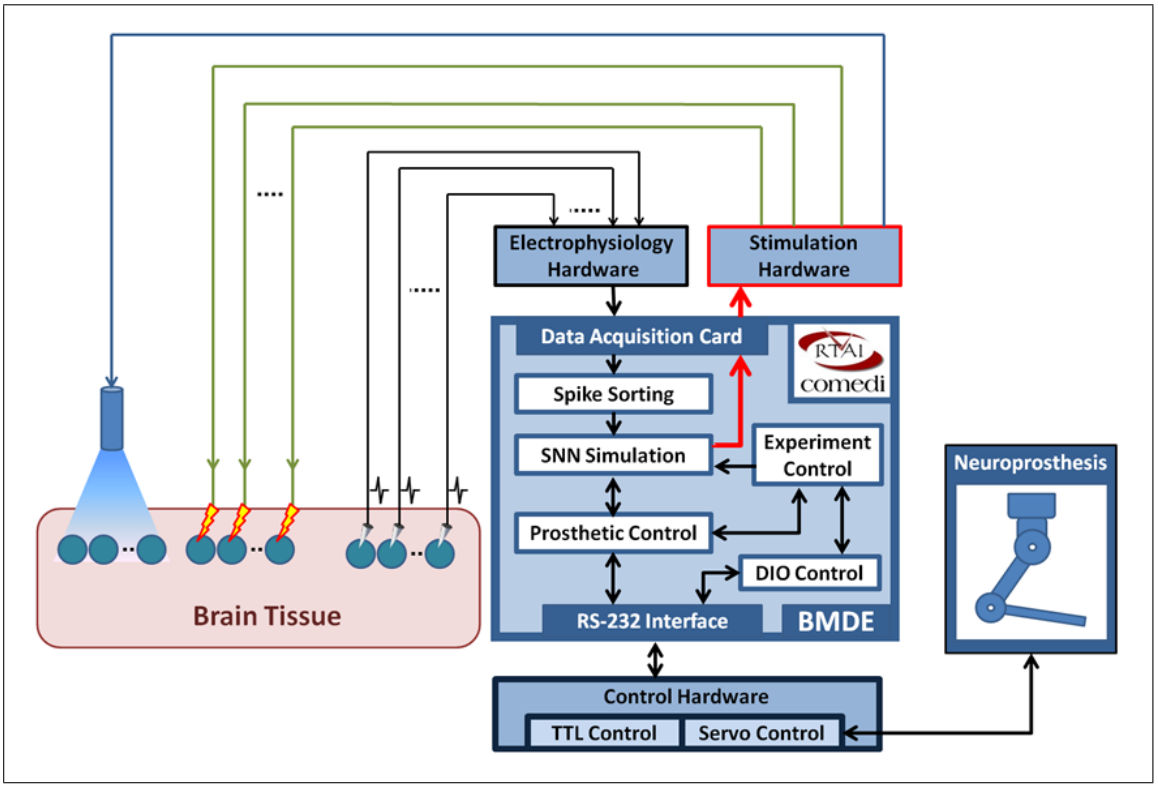

Figure 7.2 Implementation of electrical and/or optical stimulation on the BMDE for development of bi-directional neuroprosthetic systems.

APPENDIX A. MESSAGE FORMATS BETWEEN REAL-TIME TASKS OF THE BMDE

Directions of flow of messages between the real-time tasks of the BMDE are shown in Figure 3.2. In this section, the format and content of the messages between these tasks are listed.

From spike sorting task to SNN simulation task:

Microwire Array Number	Microwire Array Channel Number	Isolated Unit Number

Microwire Array Number: Microwire arrays are defined using the GUI of the BMDE and each microwire array has a number starting from 0.
Microwire Array Channel Number: Number of channels in a microwire array is also set using the system GUI. Each channel has a number starting from 0.
Isolated Unit Number: The detected neural spikes from a channel of a microwire array are sorted into up to 3 units. Each unit has a number starting from 0. The spike sorting task delivers this message to SNN simulation task whenever it extracts and sorts a spike from the neural recordings.

From SNN simulation task to prosthetic control task:

Layer Number Number	Neuron Group Number	Neuron Number

The *in silico* network in the system consists of network layers which include neuron groups formed by identical type of neurons. For instance, the *in silico* network in the corticostriatal circuit-inspired neuroprosthetic control (Chapter 4.5) consists of two network layers. Each layer comprises of one neuron group including medium spiny neuron types. Each neuron group includes one medium spiny neuron.

Whenever a spike generated in the output layer of the *in silico* network (or in other words in the SNN), the SNN simulation task delivers the above message to the prosthetic control task to be converted into prosthetic movements.

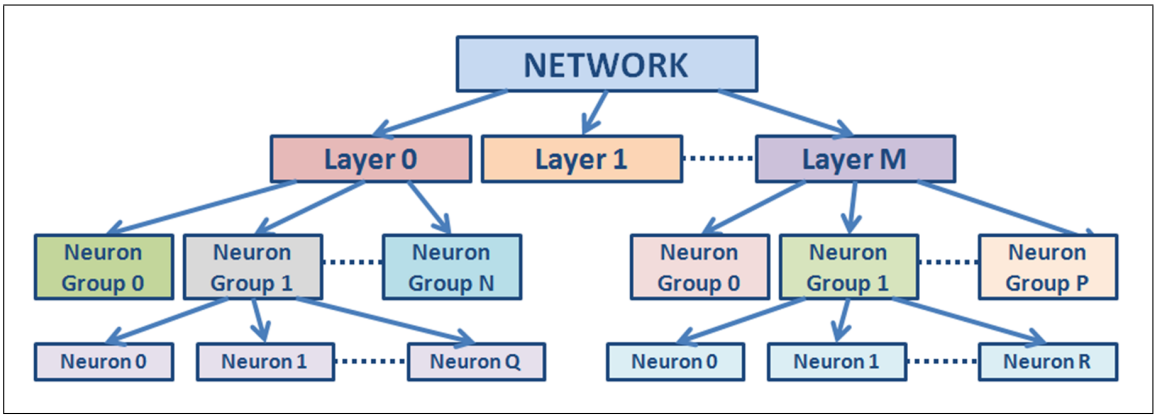

Figure A.1 Generic *in silico* neural network architecture in the BMDE. Each neuron group includes same type of neurons.

From prosthetic control task to SNN simulation task:

Message Time	Message Type	Additional Info

Message Time: Delivery time of the message in nanoseconds. It is provided by reading the system time just before forming and sending the message.
Message Type: The messages are handled by the SNN simulation task according its type. Some examples of available message types:

PROSTHETIC_CTRL_2_NEURAL_NET_MSG_REINFORCEMENT
PROSTHETIC_CTRL_2_NEURAL_NET_MSG_START_TRIAL

Additional Info: A value is also included in additional information part of the message. For instance, PROSTHETIC_CTRL_2_NEURAL_NET_MSG_REINFORCEMENT message is delivered with the sign of (negative (-1) or positive (+1)) the reinforcement.

From DIO control task to experiment control task:

Message Time	Message Type	Additional Info

Message Type: The (request) messages are handled by the experiment control task according its type and decisions are made by the experiment control task according to the experimental paradigm. Some examples of available message types:

DIO_CTRL_2_EXP_CTRL_MSG_START_TRIAL_REQUEST
DIO_CTRL_2_EXP_CTRL_MSG_REWARD_REQUEST
DIO_CTRL_2_EXP_CTRL_MSG_PUNISHMENT_REQUEST
DIO_CTRL_2_EXP_CTRL_MSG_END_TRIAL_REQUEST

Additional Info: It is placed into this message but not used currently. It can be used in future implementations with addition of new message types.

From prosthetic control task to experiment control task:

Message Time	Message Type	Additional Info

Message Time: Delivery time of the message in nanoseconds.
Message Type: The (request) messages are handled by the experiment control task according its type. Some examples of available message types:

PROSTHETIC_CTRL_2_EXP_CTRL_MSG_REWARD_REQUEST
PROSTHETIC_CTRL_2_EXP_CTRL_MSG_PUNISHMENT_REQUEST
PROSTHETIC_CTRL_2_EXP_CTRL_MSG_END_TRIAL_REQUEST
PROSTHETIC_CTRL_2_EXP_CTRL_MSG_REACHED_TARGET
PROSTHETIC_CTRL_2_EXP_CTRL_MSG_REACHED_WRONG_TARGET

Additional Info: It is placed into this message but not used currently. It can be used in future implementations with addition of new message types.

From prosthetic control task to experiment control task:

Message Time	Message Type	Additional Info

Message Time: Delivery time of the message in nanoseconds.
Message Type: The (request) messages are handled by the experiment control task according its type. Some examples of available message types:

EXP_CTRL_2_DIO_CTRL_MSG_START_TRIAL
EXP_CTRL_2_DIO_CTRL_MSG_END_TRIAL
EXP_CTRL_2_DIO_CTRL_MSG_RELEASE_REWARD
EXP_CTRL_2_DIO_CTRL_MSG_RELEASE_PUNISHMENT
EXP_CTRL_2_DIO_CTRL_MSG_START_RECORDING
EXP_CTRL_2_DIO_CTRL_MSG_STOP_RECORDING
EXP_CTRL_2_DIO_CTRL_MSG_CANCEL_RECORDING

Additional Info: It is used, for instance, with EXP_CTRL_2_DIO_CTRL_MSG_START_TRIAL message type to inform the DIO control task about the selected target

(left or right).

From experiment control task to prosthetic control task:

Message Time	Message Type	Additional Info

Message Time: Delivery time of the message in nanoseconds.
Message Type: The (request) messages are handled by the experiment control task according its type. Some examples of available message types:

EXP_CTRL_2_PROSTHETIC_CTRL_MSG_START_TRIAL
EXP_CTRL_2_PROSTHETIC_CTRL_MSG_TRIAL_TIMEOUT
EXP_CTRL_2_PROSTHETIC_CTRL_MSG_END_TRIAL
EXP_CTRL_2_PROSTHETIC_CTRL_MSG_START_RECORDING
EXP_CTRL_2_PROSTHETIC_CTRL_MSG_STOP_RECORDING
EXP_CTRL_2_PROSTHETIC_CTRL_MSG_CANCEL_RECORDING
EXP_CTRL_2_PROSTHETIC_CTRL_MSG_CHANGE_TARGET

Additional Info: It is used, for instance, with EXP_CTRL_2_PROSTHETIC_ CTRL_MSG_START_TRIAL message type to inform the prosthetic control task about the selected target (left or right) so that the prosthetic control task informs the experiment control task when it reaches correct or wrong target.

APPENDIX B. PSEUDOCODES FOR REAL-TIME TASKS OF THE BMDE

The Spike Sorting Task:

```
initialize_rt_task ();
assign_rt_task_to_cpu_0 ();
configure_daq_card (); // for continuous ADC and interrupt delivery for every 16
scans.
initiate_daq_process ();
t_task = rt_get_time_ns(); // read system time to evaluate latency in DAQ inter
rupt generation.
wait_daq_interrupt ();
t_latency = rt_get_time_ns() - t_task - 512μs; // evaluate latency in DAQ inter
rupt generation. 512us is the time to acquire 16 scans for each channel.
while (1)
{
        wait_daq_interrupt ();
        t_task = rt_get_time_ns(); // read system time to evaluate times of sorted
spikes.
        for (i=0; i<number_of_daq_channels; i++)
        {
                detect_spikes_via_level_thresholding (i);
                if (spike_detected)
                {
                        run_template_matching_for_spike_sorting();
                        evaluate_spike_time ();
                        schedule_synaptic_event_for_postsynaptic_in_silico_target
_neuron(); // deliver detected single − unit spike to the SNN simulation task.
                } } }
```

The SNN Simulation Task:

initialize_periodic_rt_task (); // *the task period is* 2 *ms.*
assign_rt_task_to_cpu_x (); // *x is* 1 *or* 2 *according to the number of SNN simulation task. There are two SNN simulation tasks running on different CPU cores.*
t = rt_get_ time_ns(); // *get current CPU time in nanoseconds.*
t_prev = t; // *keep current cpu time*
while (1)
{
    rt_task_wait_period(); // *suspend task until next period.*
    t = rt_get_ time_ns(); // *read current system time in nanoseconds.*
    handle_messages_from_experiment_control_task();
    handle_messages_from_prosthetic_control_task();
    simulate_network_by_numerical_integration (t_prev, t); // *apply Parker − Sochacki integration method between t_prev and t.*
    t_prev = t;
}

void simulate_network_by_numerical_integration (RTIME t_prev, RTIME t)
// *RTIME is of type long long int to represent system time in nanoseconds.*
{
    time = t_prev;
    while (time < t) // *integrate for a step*
    {
        sort_synaptic_events(); // *sort scheduled events both from real and in silico neurons.*
        step_end = time+delta_t; // *delta_t is the integration step size of* 0.25 *ms.*
        if (step_end >= t)
            break;

```
            HANDLE_EVENTS:
            t_event = get_next_synaptic_event_time();
            if (t_event < step_end)
            {
                  // then there is a synaptic event corresponding to the current
integration step
                  update_synaptic_conductance_values();
                  integrate_step(time, t_event) // integrate with a step − size
of t_event − time
                  time = t_event;
                  goto HANDLE_EVENTS;
            }
            integrate_step (time, step_end);
            time = step_end;
      }
      integrate_step (time,t);
}

void integrate_step (RTIME t_prev, RTIME t)
{
      ps_integrate(t_prev,t); //integration using Parker − Sochacki method for a
step between t_prev and t.
      if (v > v_peak) //spikegenerated.
      {
            spike_time = evaluate_spike_time (); // detect spike time where v =
v_peak using Newton − Raphson method.
            schedule_event_for_post_synaptic_neurons();
            if (output_neuron) // if the simulated neuron is an output layer neuron
            {
                  schedule_event_for_proshetic_control_task();
            }
            update_neuron_params(spike_time); // update u and conductance
```

values where v = v_peak.

ps_integrate(spike_time,t); // *integrate for the after − spike (remaining) part.*

}

}

The Prosthetic Control Task:

```
initialize_periodic_rt_task (); // the task period is 2 ms.
assign_rt_task_to_cpu_3 ();
while (1)
{
    rt_task_wait_period(); // suspend task until next period.
    handle_messages_from_gui_of_prosthetic_control_task (); // messages
from graphical user interface (GUI) settings prosthetic control are realized via
GUI.
    handle_messages_from_experiment_control_task (); // trial start, end
etc.
    buffer_spike_events_delivered_by_snn_simulation_task();
    if (data_transmission_time_to_control_hardware) // deliver pulse width
commands to control hardware every 26 ms.
    {
        bin_spikes_from_SNN_simulation_task ();
        determine_robotic_action_according_to_spike_counts ();
        deliver_pulse_width_command_to_control_hardware ();
    }
    if (data_receive_time_from_control_hardware) // read message from the
control hardware 10 ms after delivering the pulse width command.
    {
        read_message_received_from_control_hardware ();
        evaluate_joint_angles_and_robot_tip_position ();
        if (target_reach) // if one of the targets is acquired.
        {
            transmit_message_to_experiment_control_task ();
        }
    }
}
```

The Digital Input-Output (DIO) Control Task:

initialize_periodic_rt_task (); // *the task period is* 2 *ms.*

assign_rt_task_to_cpu_3 ();

while (1)

{

rt_task_wait_period(); // *suspend task until next period.*

handle_messages_from_gui_of_dio_control_task (); // *messages from graphical user interface (GUI). settings for experimental environment components are realized using GUI.*

handle_messages_from_experiment_control_task (); // *trial start, trial end, start data recording etc.*

determine_ binary_states_of_experiment_environment_output_components (); // *state of LEDS, solenoid valve etc.*

write_state_of_outputs_to_memory (); // *the prosthetic control task will read and deliver them to control hardware while sending its pulse width commands.*

read_message_received_from_control_hardware ();

determine_ binary_states_of_experiment_environment_input_components ();

deliver_request_message_to_experiment_control_task ();

}

The Experiment Control Task:

initialize_periodic_rt_task (); // *the task period is* 2 *ms.*
assign_rt_task_to_cpu_3 ();
while (1)
{
rt_task_wait_period(); // *suspend task until next period.*
handle_messages_from_gui_of_experiment_control_task (); // *messages from graphical user interface (GUI). settings for experimental paradigm (e.g. trial duration) and data recording initation/termination for all tasks are realized using GUI.*
handle_messages_from_dio_control_task (); // *e.g. trial start request.*
handle_messages_from_prosthetic_control_task (); // *e.g. target acquired.*
check_time_related_conditions_of_experimental_paradigm (); // *e.g. trial timeout*?
transmit_message_to_ dio_control_task (); // *e.g. trial start/end data recording start/end etc.*
transmit_message_to_ prosthetic_control_task (); // *e.g. trial start/end data recording start/end etc.*
transmit_message_to_ snn_simulation_task (); // *e.g. trial start/end data recording start/end etc.*
}

APPENDIX C. SCHEMATIC DIAGRAM OF THE CONTROL HARDWARE

The schematic diagram of the control hardware is available in the following page.

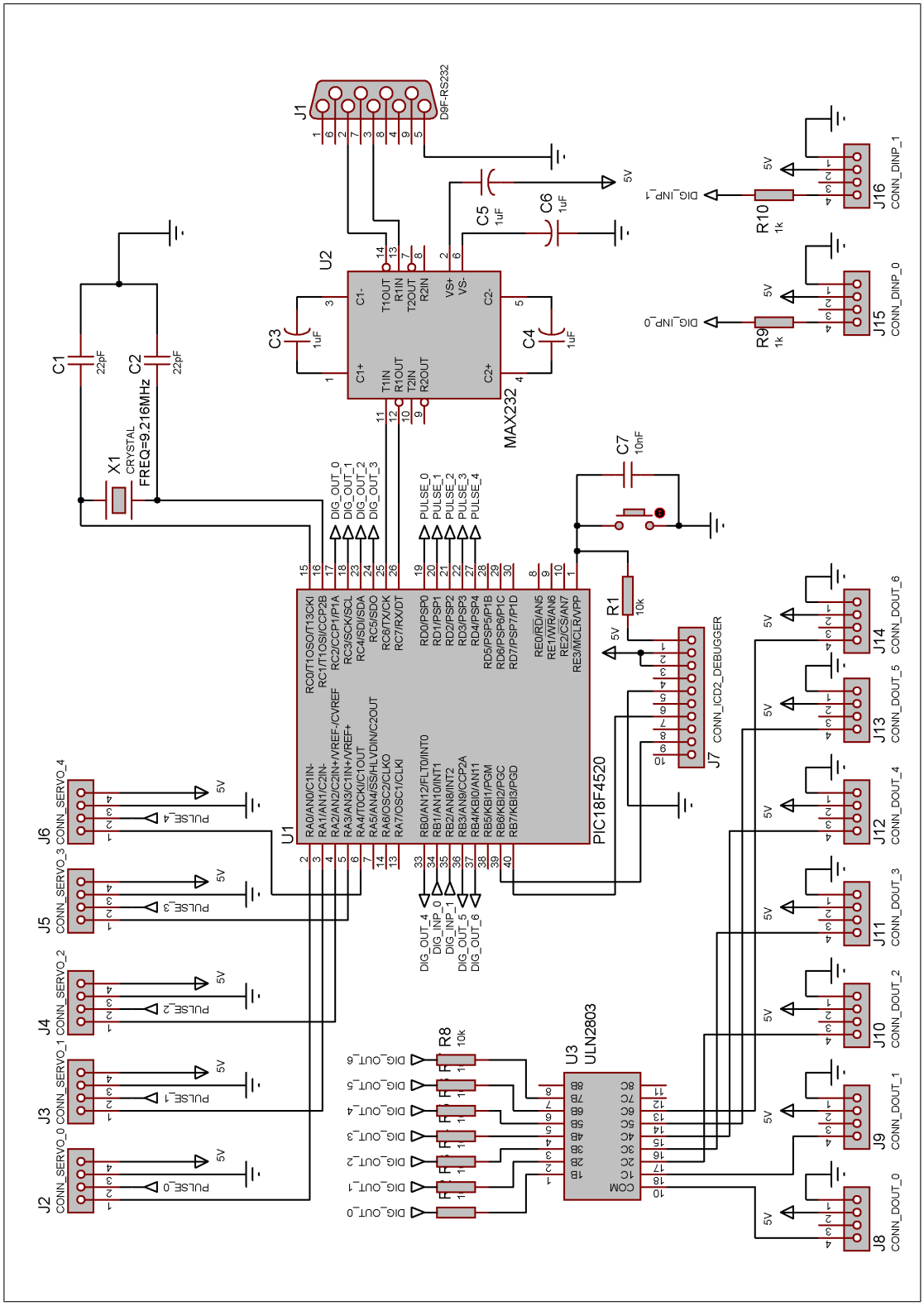

Figure C.1 Schematic diagram of the Control Hardware.

APPENDIX D. PSEUDOCODE FOR THE CONTROL HARDWARE

configure_microcontroller();
while (1)
{
wait_rx_interrupt(); // *wait for a message from the BMDE through* $RS-232$ *port message is periodically received every* 26 *ms.*
parse_message_for_servo_and_ttl_control_modules();
apply_servo_pulse_width_command_signals() ; // *drive three servo motors of the robotic arm.*
set_the_binary_state_of_experiment_environment_components(); // *turn on/off LEDS etc.*
measure_servo_angles(); // *read joint angles of the rob. arm through* $analog-to-digital$ *conversion.*
read_status_of_binary_stated_experiment_environment_components(); // *read status of levers etc.*
form_message_to_bmde();
deliver_message_to_bmde(); // *via* $RS-232$ *port*
}

APPENDIX E. GRAPHICAL USER INTERFACES OF THE BMDE

The graphical user interfaces (GUIs) of the Bioinspired Model Development Environment (BMDE) are available in the next pages.

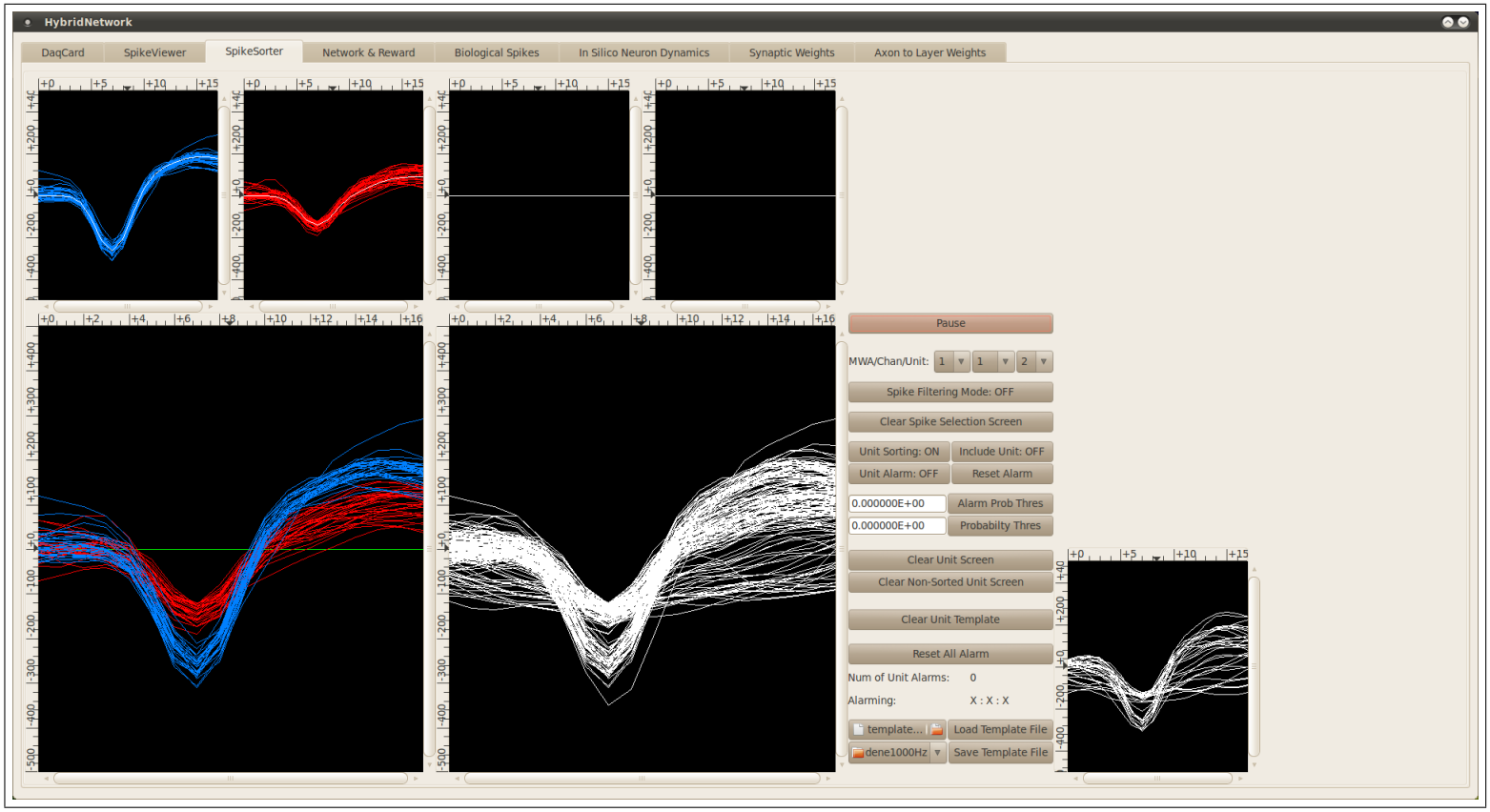

Figure E.1 Spike sorting graphical user interface. Snapshot taken during *in vivo* recording from awake rat implanted in the primary motor cortex forelimb area.

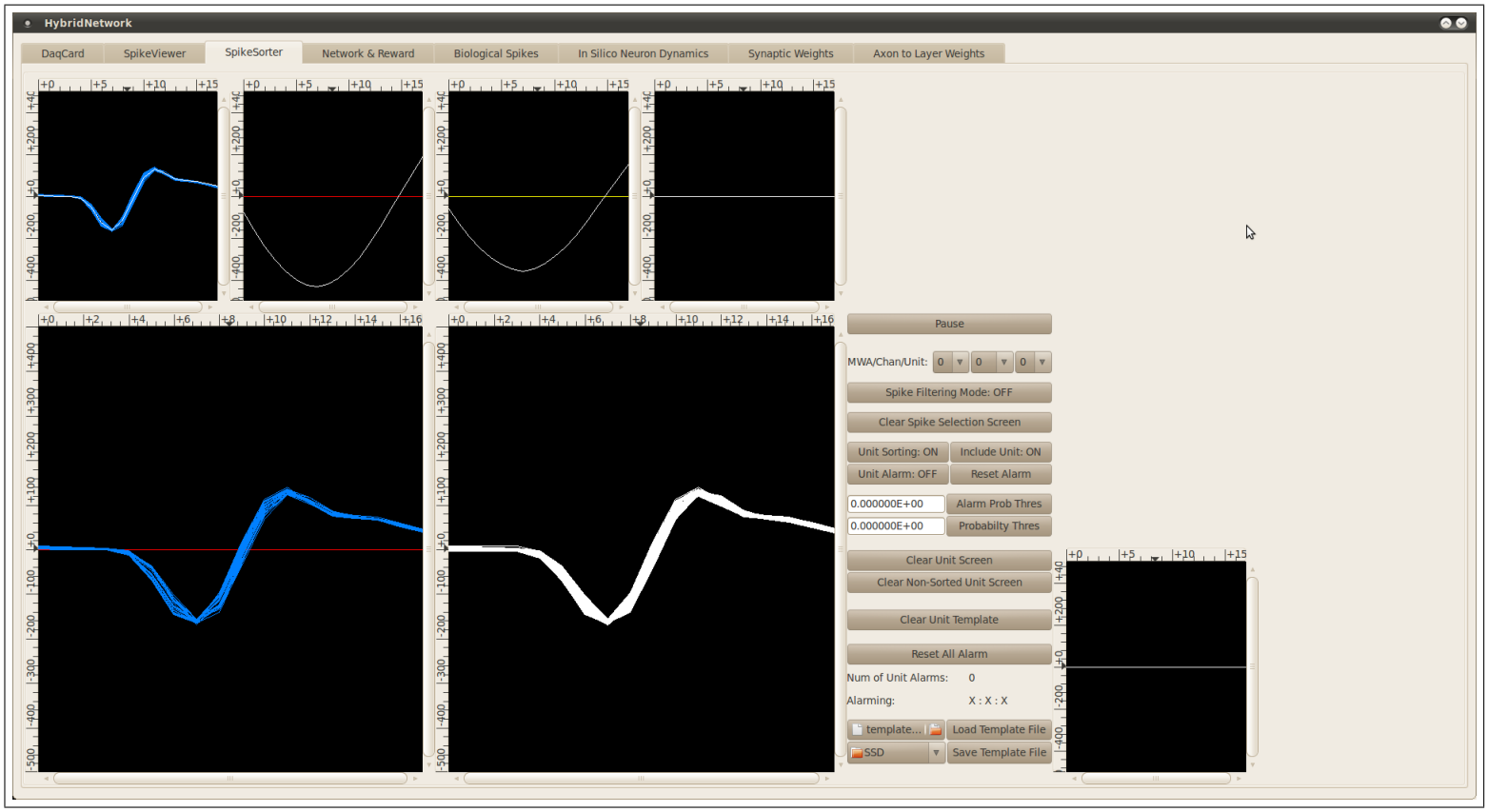

Figure E.2 Spike sorting graphical user interface. Snapshot taken during recording the outputs of the neural signal synthesizer. The first spike template was generated by selecting the spike waveforms related to neural signal synthesizer. The other two templates were generated by applying sinusoidal waveform to the analog input channels of the data acquisition hardware.

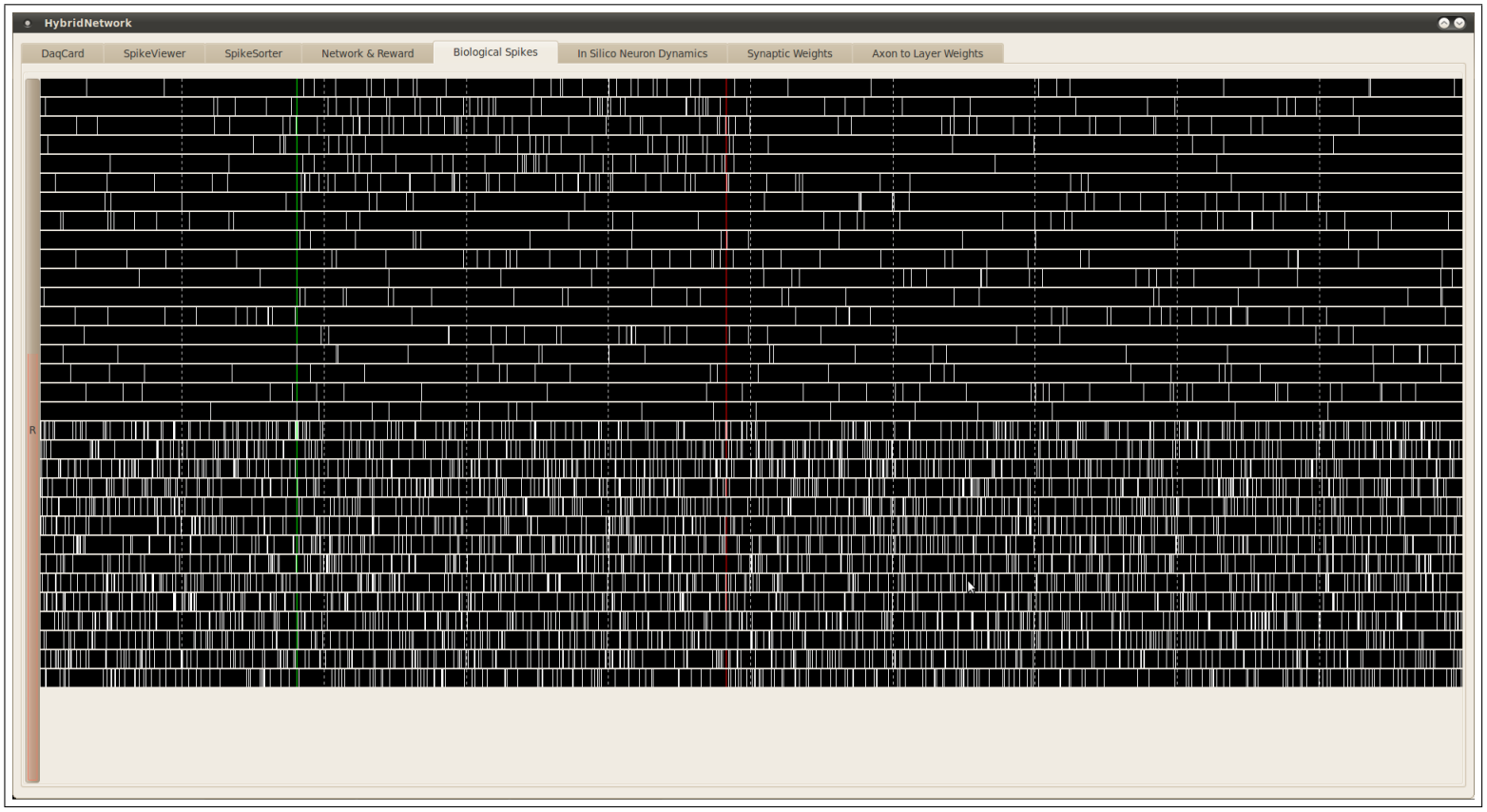

Figure E.3 Graphical user interface for monitoring raster of the spikes extracted from recordings. The snapshot taken during 462nd trial of the stress test presented in Chapter 6.1. The first six rows present the tuned activity of the neurons simulated by the Synt-A. Last 14 rows reflect the activity pattern of the high-frequency spiking neurons of the Synt-B. The green vertical line marks the timepoint on which the control of the robotic actuator was enabled and the red line indicates the timepoint on which the trial was ended by acquisition of the right target. The time interval between the vertical dashed lines is 300 milliseconds and the voltage difference between the horizontal dashed lines is 30 mV.

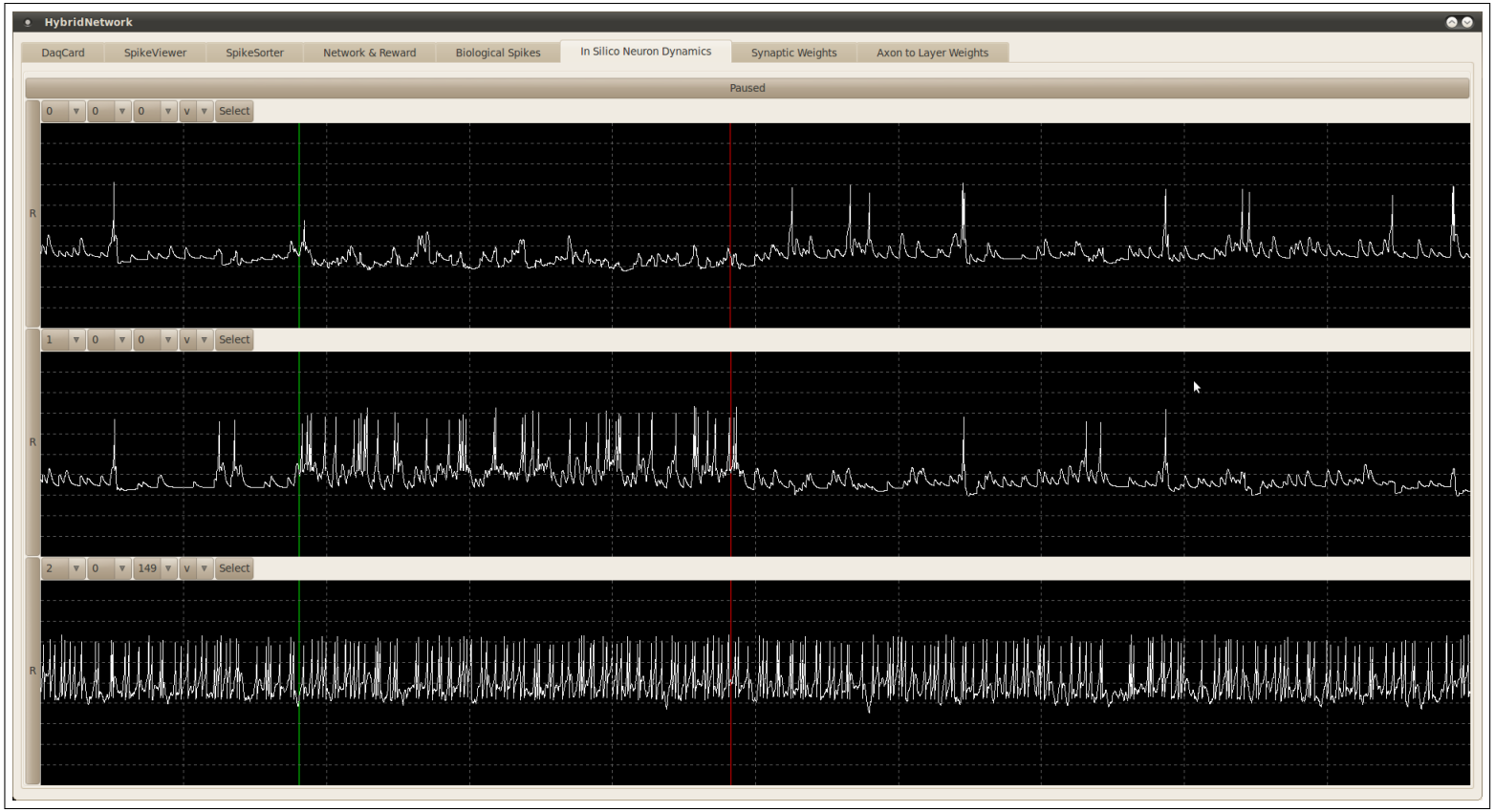

Figure E.4 Dynamics of the MSNs during 462nd trial of the stress test (presented in Chapter 6.1) in response to the spike events presented in Figure E.3. When the trial was started and the robot control was enabled (vertical green line), the MSN corresponding to right action was activated by the increasing activity of the simulated motor cortex units (graph in the middle) and suppressed the activity of the left action MSN (uppermost graph) through lateral inhibition. At the end of the trial by successful target reach (vertical red line), the MSNs returned to their baseline activities with the decrease in the activities of the presynaptic motor cortex units. The bottom graph represents the membrane dynamics of one the MSNs which were added into SNN for the stress test paradigm. The green vertical line marks the timepoint on which the control of the robotic actuator was enabled. The time interval between the vertical dashed lines is 300 milliseconds and the voltage difference between the horizontal dashed lines is 30 mV.

APPENDIX F. THE LIST OF PUBLICATIONS ORIGINATED FROM THE PRESENT DISSERTATION WORK

F.1 Publications in Journals

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., Toward building hybrid biological/in silico neural networks for motor neuroprosthetic control, Frontiers in Neurorobotics, 9:8, 2015.

F.2 International Conference Proceeding Papers

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., A workstation for development of brain machine interfaces using spiking neural networks, Proceedings of the International Conference on Neurorehabilitation (ICNR 2012), Converging Clinical and Engineering Research on Neurorehabilitation, Toledo, Spain. Pages: 623-627, 2012.

F.3 International Conference Abstracts

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., Towards brain machine interfaces operating with spike-based models of neurons, Neuroscience 2014, Washington DC, USA, 2014.

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., A real-time spiking neural network simulator combined with online spike sorting software, Neuroscience 2012, New Orleans, USA, 2012.

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., BlueSpike: Neural data acquisition, recording and online spike sorting software, Neuroscience 2011, Washington DC, USA, 2011.

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., A real-time data acquisition and neural spike processing platform for brain machine interface engineering experiments, Neuroscience 2010, San Diego, USA, 2010.

F.4 National Conference Proceeding Papers

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., Real-time neural network simulation software for brain machine interface engineering, 17th National Conference on Biomedical Engineering (BIYOMUT 2012), Istanbul, Turkey, 2012.

Serbes, G., Kocatürk, M, Gülçür, H. Ö., Aydın, N., Extracellular spike detection with resonance based signal decomposition, 20th Signal Processing and Communications Applications Conference (SIU 2012), Fethiye, Turkey, 2012.

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., BlueSpike: Online neural spike sorting software for brain machine interface engineering experiments, 16th National Conference on Biomedical Engineering (BIYOMUT 2011), Antalya, Turkey, 2011.

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., Chronic recordings from rat motor cortex for developing neural prostheses, 15th National Conference on Biomedical Engineering (BIYOMUT 2010), Antalya, Turkey, 2010.

F.5 National Conference Abstracts

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., A real-time simulation study for a biomimetic brain machine interface, 12th National Neuroscience Congress (USK 2014), Istanbul, Turkey, 2014.

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., A low-cost microelectrode array for brain-machine interfacing, 10th National Neuroscience Congress (USK 2011), Istanbul, Turkey, 2011.

Bayat, F.K., Kocatürk, M., Gülçür H. Ö., Canbeyli, R., Correlated activity decoding of motor cortical spike recordings using spike firing rate estimates, 10th National Neuroscience Congress (USK 2011), Istanbul, Turkey, 2011.

Kocatürk, M., Gülçür, H. Ö., Canbeyli, R., BlueSpike: An online data acquisition and neural spike processing platform for brain machine interface engineering experiments, 10th National Neuroscience Congress (USK 2011), Istanbul, Turkey, 2011.

REFERENCES

1. Hochberg, L. R., D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. van der Smagt, and J. P. Donoghue, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm.," *Nature*, Vol. 485, pp. 372–5, May 2012.

2. Collinger, J. L., B. Wodlinger, J. E. Downey, W. Wang, E. C. Tyler-Kabara, D. J. Weber, A. J. C. McMorland, M. Velliste, M. L. Boninger, and A. B. Schwartz, "High-performance neuroprosthetic control by an individual with tetraplegia.," *Lancet*, Vol. 381, pp. 557–64, Feb. 2013.

3. Hochberg, L. R., 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*, Vol. 442, pp. 164–71, July 2006.

4. Olds, J., "Operant conditioning of single unit responses," *Excerpta Medica International Congress Series*, Vol. 87, pp. 372–380, 1965.

5. Fetz, E. E., "Operant conditioning of cortical unit activity.," *Science (New York, N.Y.)*, Vol. 163, pp. 955–8, Feb. 1969.

6. Georgopoulos, A. P., J. F. Kalaska, R. Caminiti, and J. T. Massey, "On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 2, pp. 1527–37, Nov. 1982.

7. Georgopoulos, A. P., R. Caminiti, J. F. Kalaska, and J. T. Massey, "Spatial coding of movement: A hypothesis concerning the coding of movement direction by motor cortical populations," *Experimental Brain Research*, Vol. 49, pp. 327–336, 1983.

8. Georgopoulos, A. P., R. E. Kettner, and A. B. Schwartz, "Primate motor cortex and free arm movements to visual targets in three-dimensional space. II. Coding of the direction of movement by a neuronal population.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 8, pp. 2928–37, Aug. 1988.

9. Chapin, J. K., K. a. Moxon, R. S. Markowitz, and M. A. L. Nicolelis, "Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex.," *Nature neuroscience*, Vol. 2, pp. 664–70, July 1999.

10. Taylor, D. M., S. I. H. Tillery, and A. B. Schwartz, "Direct cortical control of 3D neuroprosthetic devices.," *Science (New York, N.Y.)*, Vol. 296, pp. 1829–32, June 2002.

11. Carmena, J. M., M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis, "Learning to Control a Brain Machine Interface for Reaching and Grasping by Primates," *PLoS Biology*, Vol. 1, no. 2, p. E42, 2003.

12. Wu, S.-t., and M.-l. Tsai, "A SIMPLE METHOD FOR FABRICATING MULTI-ROW MICROWIRE ARRAYS," *Compare A Journal Of Comparative Education*, pp. 281–282, 2004.

13. Santhanam, G., B. M. Yu, V. Gilja, S. I. Ryu, A. Afshar, M. Sahani, and K. V. Shenoy, "Factor-analysis methods for higher-performance neural prostheses.," *Journal of neurophysiology*, Vol. 102, pp. 1315–30, Aug. 2009.

14. Brockwell, A. E., A. L. Rojas, and R. E. Kass, "Recursive bayesian decoding of motor cortical signals by particle filtering.," *Journal of neurophysiology*, Vol. 91, pp. 1899–907, Apr. 2004.

15. Sanchez, J. C., and J. C. Principe, *Brain-Machine Interface Engineering*, Vol. 2, Morgan & Claypool Publishers, Jan. 2007.

16. Kleim, J. a., S. Barbay, and R. J. Nudo, "Functional reorganization of the rat motor cortex following motor skill learning.," *Journal of neurophysiology*, Vol. 80, pp. 3321–5, Dec. 1998.

17. Monfils, M.-H., P. M. VandenBerg, J. a. Kleim, and G. C. Teskey, "Long-term potentiation induces expanded movement representations and dendritic hypertrophy in layer V of rat sensorimotor neocortex.," *Cerebral cortex (New York, N.Y. : 1991)*, Vol. 14, pp. 586–93, May 2004.

18. Monfils, M.-H., and G. C. Teskey, "Skilled-learning-induced potentiation in rat sensorimotor cortex: a transient form of behavioural long-term potentiation.," *Neuroscience*, Vol. 125, pp. 329–36, Jan. 2004.

19. Jarosiewicz, B., S. M. Chase, G. W. Fraser, M. Velliste, R. E. Kass, and A. B. Schwartz, "Functional network reorganization during learning in a brain-computer interface paradigm.," *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 105, pp. 19486–91, Dec. 2008.

20. Ganguly, K., D. F. Dimitrov, J. D. Wallis, and J. M. Carmena, "Reversible large-scale modification of cortical networks during neuroprosthetic control.," *Nature neuroscience*, Vol. 14, pp. 662–7, May 2011.

21. Shenoy, K. V., and J. M. Carmena, "Combining Decoder Design and Neural Adaptation in Brain-Machine Interfaces.," *Neuron*, Vol. 84, pp. 665–680, Nov. 2014.

22. Li, Z., J. E. O'Doherty, M. A. Lebedev, and M. A. L. Nicolelis, "Adaptive decoding for brain-machine interfaces through Bayesian parameter updates.," *Neural computation*, Vol. 23, pp. 3162–204, Dec. 2011.

23. Gilja, V., P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, M. M. Churchland, M. T. Kaufman, J. C. Kao, S. I. Ryu, and K. V. Shenoy, "A high-performance neural prosthesis enabled by control algorithm design.," *Nature neuroscience*, Vol. 15, pp. 1752–7, Dec. 2012.

24. Orsborn, A. L., S. Dangi, H. G. Moorman, and J. M. Carmena, "Closed-loop decoder adaptation on intermediate time-scales facilitates rapid BMI performance improvements independent of decoder initialization conditions.," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, Vol. 20, pp. 468–77, July 2012.

25. Mahmoudi, B., E. A. Pohlmeyer, N. W. Prins, S. Geng, and J. C. Sanchez, "Towards autonomous neuroprosthetic control using Hebbian reinforcement learning.," *Journal of neural engineering*, Vol. 10, p. 066005, Oct. 2013.

26. Pohlmeyer, E. A., B. Mahmoudi, S. Geng, N. W. Prins, and J. C. Sanchez, "Using reinforcement learning to provide stable brain-machine interface control despite neural input reorganization.," *PloS one*, Vol. 9, p. e87253, Jan. 2014.

27. Velliste, M., S. D. Kennedy, A. B. Schwartz, A. S. Whitford, J.-W. Sohn, and A. J. C. McMorland, "Motor cortical correlates of arm resting in the context of a reaching task and implications for prosthetic control.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 34, pp. 6011–22, Apr. 2014.

28. Baranauskas, G., "What limits the performance of current invasive brain machine interfaces?," *Frontiers in systems neuroscience*, Vol. 8, p. 68, Jan. 2014.

29. Tehovnik, E. J., L. C. Woods, and W. M. Slocum, "Transfer of information by BMI.," *Neuroscience*, Vol. 255, pp. 134–46, Jan. 2013.

30. Markram, H., W. Gerstner, and P. J. Sjöström, "A history of spike-timing-dependent plasticity," *Frontiers in Synaptic Neuroscience*, Vol. 3, no. AUG, pp. 1–24, 2011.

31. Koralek, A. C., X. Jin, J. D. Long, R. M. Costa, and J. M. Carmena, "Corticostriatal plasticity is necessary for learning intentional neuroprosthetic skills.," *Nature*, Vol. 483, pp. 331–5, Mar. 2012.

32. Indiveri, G., B. Linares-Barranco, T. J. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen, "Neuromorphic silicon neuron circuits.," *Frontiers in neuroscience*, Vol. 5, p. 73, Jan. 2011.

33. Dethier, J., S. Member, V. Gilja, P. Nuyujukian, S. A. Elassaad, K. V. Shenoy, S. Member, and K. Boahen, "Spiking Neural Network Decoder for Brain-Machine Interfaces," no. 3, pp. 396–399, 2011.

34. Yu, Y.-m., and R.-c. Lo, "Cortical signal recording using an economical microelectrode fabricated on printed circuit board," *WSEAS Transactions on Biology and Biomedicine*, Vol. 5, no. 8, pp. 183–188, 2008.

35. Anderson, C. T., P. L. Sheets, T. Kiritani, and G. M. G. Shepherd, "Sublayer-specific microcircuits of corticospinal and corticostriatal neurons in motor cortex.," *Nature neuroscience*, Vol. 13, pp. 739–44, June 2010.

36. Gioanni, Y., and M. Lamarche, "A reappraisal of rat motor cortex organization by intracortical microstimulation.," Sept. 1985.

37. Skoglund, T. S., R. Pascher, and C. H. Berthold, "The existence of a layer IV in the rat motor cortex.," *Cerebral cortex (New York, N.Y. : 1991)*, Vol. 7, pp. 178–80, Mar. 1997.

38. DiGiovanna, J., B. Mahmoudi, J. Fortes, J. C. Principe, and J. C. Sanchez, "Coadaptive brain-machine interface via reinforcement learning.," *IEEE transactions on bio-medical engineering*, Vol. 56, pp. 54–64, Jan. 2009.

39. Prasad, A., Q.-S. Xue, R. Dieme, V. Sankar, R. C. Mayrand, T. Nishida, W. J. Streit, and J. C. Sanchez, "Abiotic-biotic characterization of Pt/Ir microelectrode arrays in chronic implants.," *Frontiers in neuroengineering*, Vol. 7, p. 2, Jan. 2014.

40. Mantegazza, P., E. Bianchi, L. Dozio, S. Papacharalambous, S. Hughes, and D. Beal, "RTAI: Real-Time Application Interface," *Linux Journal*, 2000.

41. Lewicki, M. S., "A review of methods for spike sorting: the detection and classification of neural action potentials.," *Network: Computation in Neural Systems*, Vol. 9, pp. R53–78, Nov. 1998.

42. Alpaydin, E., *Introduction to machine learning*, Cambridge, Mass.: MIT Press, 2 ed., 2010.

43. Izhikevich, E. M., "Simple model of spiking neurons," *IEEE Transactions on Neural Networks*, Vol. 14, no. 6, pp. 1569–1572, 2003.

44. Izhikevich, E. M., *Dynamical systems in neuroscience: The geometry of excitability and bursting.*, Cambridge: MIT, 2007.

45. Izhikevich, E. M., "Which Model to Use for Cortical Spiking Neurons?," *IEEE Transactions on Neural Networks*, Vol. 15, pp. 1063–1070, Sept. 2004.

46. Parker, G. E., and J. S. Sochacki, "A Picard-Maclaurin theorem for initial value PDEs," *Abstract and Applied Analysis*, Vol. 5, no. 1, pp. 47–63, 2000.

47. Stewart, R. D., and W. Bair, "Spiking neural network simulation: numerical integration with the Parker-Sochacki method.," *Journal of computational neuroscience*, Vol. 27, pp. 115–33, Aug. 2009.

48. Murray, R. M., Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, Boca Raton: CRC Press, 1994.

49. Wickens, J. R., J. N. Reynolds, and B. I. Hyland, "Neural mechanisms of reward-related motor learning," *Current Opinion in Neurobiology*, Vol. 13, pp. 685–690, Dec. 2003.

50. Turner, R. S., and M. Desmurget, "Basal ganglia contributions to motor control: A vigorous tutor," *Current Opinion in Neurobiology*, Vol. 20, no. 6, pp. 704–716, 2010.

51. Grillner, S., J. Hellgren, A. Ménard, K. Saitoh, and M. a. Wikström, "Mechanisms for selection of basic motor programs - Roles for the striatum and pallidum," *Trends in Neurosciences*, Vol. 28, no. 7, pp. 364–370, 2005.

52. Barrett, K. E., S. M. Barman, S. Boitano, and H. L. Brooks, *Ganong's Review of Medical Physiology*, New York: McGraw-Hill, 23 ed., 2010.

53. Guyton, A. C., and J. E. Hall, *Textbook of Medical Physiology*, Philadelphia: Saunders, 11 ed., 2006.

54. Scott, S. H., "Optimal feedback control and the neural basis of volitional motor control.," *Nature reviews. Neuroscience*, Vol. 5, pp. 532–46, July 2004.

55. Tepper, J. M., E. D. Abercrombie, and J. P. Bolam, "Basal ganglia macrocircuits.," *Progress in brain research*, Vol. 160, pp. 3–7, Jan. 2007.

56. Wilson, C. J., "GABAergic inhibition in the neostriatum.," *Progress in brain research*, Vol. 160, pp. 91–110, Jan. 2007.

57. Tepper, J. M., T. Koós, and C. J. Wilson, "GABAergic microcircuits in the neostriatum.," *Trends in neurosciences*, Vol. 27, pp. 662–9, Nov. 2004.

58. DeLong, M. R., and T. Wichmann, "Circuits and circuit disorders of the basal ganglia.," *Archives of neurology*, Vol. 64, pp. 20–4, Jan. 2007.

59. Kravitz, A. V., and A. C. Kreitzer, "Striatal mechanisms underlying movement, reinforcement, and punishment.," *Physiology (Bethesda, Md.)*, Vol. 27, pp. 167–77, June 2012.

60. Pisani, A., D. Centonze, G. Bernardi, and P. Calabresi, "Striatal synaptic plasticity: implications for motor learning and Parkinson's disease.," *Movement disorders : official journal of the Movement Disorder Society*, Vol. 20, pp. 395–402, Apr. 2005.

61. Yin, H. H., S. P. Mulcare, M. R. F. Hilário, E. Clouse, T. Holloway, M. I. Davis, A. C. Hansson, D. M. Lovinger, and R. M. Costa, "Dynamic reorganization of striatal circuits during the acquisition and consolidation of a skill.," *Nature neuroscience*, Vol. 12, pp. 333–41, Mar. 2009.

62. Reynolds, J. N., and J. R. Wickens, "Dopamine-dependent plasticity of corticostriatal synapses.," *Neural networks : the official journal of the International Neural Network Society*, Vol. 15, no. 4-6, pp. 507–21, 2002.

63. Kreitzer, A. C., and R. C. Malenka, "Striatal plasticity and basal ganglia circuit function.," *Neuron*, Vol. 60, pp. 543–54, Nov. 2008.

64. Pawlak, V., and J. N. D. Kerr, "Dopamine receptor activation is required for corticostriatal spike-timing-dependent plasticity.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 28, pp. 2435–46, Mar. 2008.

65. Shen, W., M. Flajolet, P. Greengard, and D. J. Surmeier, "Dichotomous dopaminergic control of striatal synaptic plasticity.," *Science (New York, N.Y.)*, Vol. 321, pp. 848–851, 2008.

66. Hyman, S. E., and R. C. Malenka, "Addiction and the brain: the neurobiology of compulsion and its persistence.," *Nature reviews. Neuroscience*, Vol. 2, pp. 695–703, Oct. 2001.

67. Reynolds, J. N., B. I. Hyland, and J. R. Wickens, "A cellular mechanism of reward-related learning.," *Nature*, Vol. 413, pp. 67–70, Sept. 2001.

68. Schultz, W., "A Neural Substrate of Prediction and Reward," *Science*, Vol. 275, pp. 1593–1599, Mar. 1997.

69. Schultz, W., "Predictive reward signal of dopamine neurons.," *Journal of Neurophysiology*, Vol. 80, no. 1, pp. 1–27, 1998.

70. Schultz, W., "Activity of pars reticulata neurons of monkey substantia nigra in relation to motor, sensory, and complex events.," *Journal of neurophysiology*, Vol. 55, pp. 660–77, Apr. 1986.

71. Hyland, B. I., J. Reynolds, J. Hay, C. Perk, and R. Miller, "Firing modes of midbrain dopamine cells in the freely moving rat," *Neuroscience*, Vol. 114, pp. 475–492, Oct. 2002.

72. Beeler, J. a., Z. F. H. Cao, M. a. Kheirbek, Y. Ding, J. Koranda, M. Murakami, U. J. Kang, and X. Zhuang, "Dopamine-dependent motor learning: insight into levodopa's long-duration response.," *Annals of neurology*, Vol. 67, pp. 639–47, May 2010.

73. Ericsson, J., M. Stephenson-Jones, J. Pérez-Fernández, B. Robertson, G. Silberberg, and S. Grillner, "Dopamine differentially modulates the excitability of striatal neurons of the direct and indirect pathways in lamprey.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 33, no. 18, pp. 8045–54, 2013.

74. Mao, Z.-h., and S. G. Massaquoi, "Dynamics of winner-take-all competition in recurrent neural networks with lateral inhibition.," *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, Vol. 18, pp. 55–69, Jan. 2007.

75. Plenz, D., and S. Kitai, "Adaptive classification of cortical input to the striatum by competitive learning," in *Brain Dynamics and the Striatal Complex* (Miller, R., and J. Wickens, eds.), ch. Adaptive c, pp. 165–178, Amsterdam: Harwood Academic, 2000.

76. O'Reilly, R. C., and M. J. Frank, "Making working memory work: a computational model of learning in the prefrontal cortex and basal ganglia.," *Neural computation*, Vol. 18, pp. 283–328, Feb. 2006.

77. Vogels, T. P., and L. F. Abbott, "Signal propagation and logic gating in networks of integrate-and-fire neurons.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 25, pp. 10786–95, Nov. 2005.

78. Frémaux, N., H. Sprekeler, and W. Gerstner, "Functional requirements for reward-modulated spike-timing-dependent plasticity.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 30, pp. 13326–37, Oct. 2010.

79. Dura-Bernal, S., G. L. Chadderdon, S. A. Neymotin, J. T. Francis, and W. W. Lytton, "Towards a real-time interface between a biomimetic model of sensorimotor cortex and a robotic arm," *Pattern Recognition Letters*, Vol. 36, pp. 204–212, Jan. 2014.

80. Eugene M. Izhikevich, "Solving the distal reward problem through linkage of STDP and dopamine signaling.," *Cerebral Cortex*, Vol. 17, no. 10, pp. 2443–2452, 2007.

81. Chadderdon, G. L., S. A. Neymotin, C. C. Kerr, and W. W. Lytton, "Reinforcement learning of targeted movement in a spiking neuronal model of motor cortex.," *PloS one*, Vol. 7, p. e47251, Jan. 2012.

82. Neymotin, S. A., G. L. Chadderdon, C. C. Kerr, J. T. Francis, and W. W. Lytton, "Reinforcement Learning of Two-Joint Virtual Arm Reaching in a Computer Model of Sensorimotor Cortex," Vol. 3293, pp. 3263–3293, 2013.

83. Vasilaki, E., N. Frémaux, R. Urbanczik, W. Senn, and W. Gerstner, "Spike-based reinforcement learning in continuous state and action space: when policy gradient methods fail.," *PLoS computational biology*, Vol. 5, p. e1000586, Dec. 2009.

84. Turrigiano, G. G., "Homeostatic plasticity in neuronal networks: the more things change, the more they stay the same," *Trends in Neurosciences*, Vol. 22, pp. 221–227, May 1999.

85. Royer, S., and D. Paré, "Conservation of total synaptic weight through balanced synaptic depression and potentiation.," *Nature*, Vol. 422, pp. 518–22, Apr. 2003.

86. Abbott, L. F., and S. B. Nelson, "Synaptic plasticity: taming the beast.," *Nature neuroscience*, Vol. 3 Suppl, pp. 1178–83, Nov. 2000.

87. Mahmoudi, B., J. C. Principe, and J. C. Sanchez, "An Actor-Critic architecture and simulator for goal-directed Brain-Machine Interfaces.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, Vol. 2009, pp. 3365–8, Jan. 2009.

88. Mahmoudi, B., and J. C. Sanchez, "A symbiotic brain-machine interface through value-based decision making.," *PloS one*, Vol. 6, p. e14760, Jan. 2011.

89. Sanchez, J. C., J. M. Carmena, M. A. Lebedev, M. A. L. Nicolelis, J. G. Harris, and J. C. Principe, "Ascertaining the importance of neurons to develop better brain-machine interfaces.," *IEEE transactions on bio-medical engineering*, Vol. 51, pp. 943–53, June 2004.

90. Wahnoun, R., J. He, and S. I. Helms Tillery, "Selection and parameterization of cortical neurons for neuroprosthetic control.," *Journal of neural engineering*, Vol. 3, pp. 162–71, June 2006.

91. Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, Vol. 29, Cambridge: Cambridge University, 2 ed., Nov. 1992.

92. Schultz, W., "Reward signaling by dopamine neurons.," *The Neuroscientist : a review journal bringing neurobiology, neurology and psychiatry*, Vol. 7, pp. 293–302, Aug. 2001.

93. Tobler, P. N., C. D. Fiorillo, and W. Schultz, "Adaptive coding of reward value by dopamine neurons.," *Science*, Vol. 307, no. 5715, pp. 1642–1645, 2005.

94. Putrino, D., E. N. Brown, F. L. Mastaglia, and S. Ghosh, "Differential involvement of excitatory and inhibitory neurons of cat motor cortex in coincident spike activity related to behavioral context.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 30, pp. 8048–56, June 2010.

95. Fee, M. S., P. P. Mitra, and D. Kleinfeld, "Variability of extracellular spike waveforms of cortical neurons.," *Journal of neurophysiology*, Vol. 76, pp. 3823–33, Dec. 1996.

96. Arduin, P. J., Y. Frégnac, D. E. Shulz, and V. Ego-Stengel, "Bidirectional control of a one-dimensional robotic actuator by operant conditioning of a single unit in rat motor cortex.," *Frontiers in neuroscience*, Vol. 8, p. 206, Jan. 2014.

97. Plenz, D., "When inhibition goes incognito: Feedback interaction between spiny projection neurons in striatal function," *Trends in Neurosciences*, Vol. 26, no. 8, pp. 436–443, 2003.

98. Koós, T., and J. M. Tepper, "Inhibitory control of neostriatal projection neurons by GABAergic interneurons.," *Nature neuroscience*, Vol. 2, no. 5, pp. 467–472, 1999.

99. Gerfen, C. R., "Molecular effects of dopamine on striatal-projection pathways.," *Trends in neurosciences*, Vol. 23, no. 10 Suppl, pp. S64–S70, 2000.

100. Humphries, M. D., R. D. Stewart, and K. N. Gurney, "A physiologically plausible model of action selection and oscillatory activity in the basal ganglia.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, Vol. 26, pp. 12921–42, Dec. 2006.

101. Stewart, T. C., T. Bekolay, and C. Eliasmith, "Learning to select actions with spiking neurons in the Basal Ganglia.," *Frontiers in neuroscience*, Vol. 6, p. 2, Jan. 2012.

102. Riehle, A., "Spike Synchronization and Rate Modulation Differentially Involved in Motor Cortical Function," *Science*, Vol. 278, pp. 1950–1953, Dec. 1997.

103. Grammont, F., and A. Riehle, "Spike synchronization and firing rate in a population of motor cortical neurons in relation to movement direction and reaction time.," *Biological cybernetics*, Vol. 88, pp. 360–73, May 2003.

104. Hatsopoulos, N. G., C. L. Ojakangas, L. Paninski, and J. P. Donoghue, "Information about movement direction obtained from synchronous activity of motor cortical neurons.," *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 95, pp. 15706–11, Dec. 1998.

105. Grammont, F., and a. Riehle, "Precise spike synchronization in monkey motor cortex involved in preparation for movement.," *Experimental brain research. Experimentelle Hirnforschung. Expérimentation cérébrale*, Vol. 128, pp. 118–22, Sept. 1999.

106. Engelhard, B., N. Ozeri, Z. Israel, H. Bergman, and E. Vaadia, "Inducing γ oscillations and precise spike synchrony by operant conditioning via brain-machine interface.," *Neuron*, Vol. 77, pp. 361–75, Jan. 2013.

107. Clark, J. J., S. G. Sandberg, M. J. Wanat, J. O. Gan, E. A. Horne, A. S. Hart, C. A. Akers, J. G. Parker, I. Willuhn, V. Martinez, S. B. Evans, N. Stella, and P. E. M. Phillips, "Chronic microsensors for longitudinal, subsecond dopamine detection in behaving animals.," *Nature methods*, Vol. 7, pp. 126–9, Feb. 2010.

108. Han, X., X. Qian, J. G. Bernstein, H.-H. Zhou, G. T. Franzesi, P. Stern, R. T. Bronson, A. M. Graybiel, R. Desimone, and E. S. Boyden, "Millisecond-timescale optical control of neural dynamics in the nonhuman primate brain.," *Neuron*, Vol. 62, pp. 191–8, Apr. 2009.

109. Venkatraman, S., and J. M. Carmena, "Active sensing of target location encoded by cortical microstimulation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 19, no. 3, pp. 317–324, 2011.