Temporal information processing and memory guided behaviors with recurrent neural networks

Dissertation
for the award of the degree
"Doctor rerum naturalium"
of the Georg-August-Universität Göttingen

within the doctoral program Physics of Biological and Complex Systems (PBCS)
of the Georg-August University School of Science (GAUSS)

submitted by
Sakyasingha Dasgupta

from Kolkata, India (place of origin)
Göttingen, 2014

Thesis committee

Prof. Dr. Florentin Wörgötter,
Biophysik - III. Physikalisches Institut, Georg-August-Universität Göttingen

Prof. Dr. Marc Timme,
Network Dynamics, Max Planck Institut für Dynamik und Selbstorganisation, Göttingen

Prof. Dr. Poramate Manoonpong,
Center for Biorobotics, Maersk-Moller Institute, Southern Denmark University, Odense

Members of the examination board

First Reviewer: **Prof. Dr. Florentin Wörgötter**,
Biophysik - III. Physikalisches Institut, Georg-August-Universität Göttingen

Second Reviewer: **Prof. Dr. Poramate Manoonpong**,
Center for Biorobotics, Maersk-Moller Institute, Southern Denmark University, Odense

Other members of the examination board:

Prof. Dr. Marc Timme,
Network Dynamics, Max Planck Institut für Dynamik und Selbstorganisation, Göttingen

Prof. Dr. Theo Geisel,
Department of Non-linear Dynamics, Max Planck Institut für Dynamik und Selbstorganisation, Göttingen

Prof. Dr. Jörg Enderlein,
Biophysik - III. Physikalisches Institut, Georg-August-Universität Göttingen

Prof. Dr. Ulrich Parlitz,
Complex Dynamical Systems, Max Planck Institut für Dynamik und Selbstorganisation, Göttingen

Date of the oral examination: 28th of January, 2015

”Nature uses only the longest threads to weave her patterns, so that each small piece of her fabric reveals the organization of the entire tapestry.”

Richard. P. Feynman

This thesis is dedicated to my father Swapan, to my mother Jayshree,
and to Richard P. Feynman, whose writings and scientific ideas have had a profound influence on my journey in science.

Acknowledgements

This thesis would not have been possible without the help, support, guidance and friendship of numerous individuals. First and foremost, I would like to thank Prof. Dr. Florentin Wörgötter for accepting me as a doctoral student in his Computational Neuroscience Lab. Over the last three years, he has not only provided me with valuable scientific guidance, but has also shaped my way of thinking. It is with my interactions with him that I learned to juxtapose the intricate neural mechanisms in the brain with resultant cognitive behaviors. I would also like to thank him for teaching me how to skii. I would like to thank Prof. Dr. Poramate Manoonpong for allowing me to be a part of his Emmy Noether Research group in Göttingen, and for supporting me at every step and also for accepting my scientific ideas. I have learned a great deal from him, specially in the field of neuro-robotics and control. Without his guidance and support, non of the robot experiments in this thesis and our subsequent publications, would have been possible. I am also thankful to him for letting me supervise a number of masters and bachelor thesis, which eventually led to successful publications. This has helped me learn a great deal about scientific collaborations as well as critically evaluate my own research. I would also like to thank Prof. Dr. Marc Timme for providing me crucial feedback during our thesis committee meetings and also help shape my research direction.

During the last three years of my life in the Wörgötter and Manoonpong labs I have met a number of amazing individuals. Many of whom I have now come to know as good friends. I would like to thank the whole group, especially Christian Tetzlaff, Xiaofeng Xiong, Dennis Goldschmidt, Tomas Kulvicius, Jeremie Papon, Christoph Kolodziejski, Alexey Abramov, Jan-Matthias Braun, Eren Erdal Aksoy, Yinyun Li and Minija Tamosiunaite. I have also had the opportunity to collaborate and interact with a number of extraordinary people outside the lab. In this regard, I would like to thank Dr. Joseph Lizier from CSIRO, Sydney Australia, for providing me with the code to the Java Information Dynamics Toolkit, which was used and extended to a great degree for all the information theoretic measures presented in this thesis. I would also like to thank Dr. Jun Morimoto from the ATR Computational Neuroscience Labs, Kyoto Japan, for collaborating with me on the topic of actor-critic reinforcement learning. Special thanks also goes to Prof. Dr. Michael Wibral from the Brain Imaging Center, Frankfurt, as well as Guanjiao Ren from Lenovo Research China and Yuichi Ambe from the University of Kyoto. Last but not the least, I would also like to thank Dr. Tomoki Fukai, Prof. Shuni-ichi Amari and Dr. Taro Toyoizumi from the Riken Brain Science Institute Japan, for appreciating my research ideas and providing me with the opportunity to continue further development of my work under their guidance.

I should also take the opportunity to appreciate the effort and support provided by our department secretary Ursula Hahn Wörgötter and the IMPRS PBCS program co-ordinator Antje Erdmann. As an international student one faces many difficulties settling down in a foreign city. Over the last four years the friendship of many individuals has made it a truly remarkable experience. I would specially like to mention, Theresa Wollenberg, Karthik Peddireddy, Devranjan Samanta, Benno Schubert, Gabriel Ducatti, David Hofmann, Mirko Lucovic, Markus Helmer and Dominika Lyzwa. I thank my family for their constant words of encouragement and love,

despite of being half a world away. I could have never made it here without the unwavering support of my father, Swapan, and my mother, Jayshree. Finally, I would like to thank Alana for being the greatest source of inspiration in my life and for being patient, understanding and for the unconditional love through all those long hours of work.

Thank you all very much indeed !
— Sakyasingha Dasgupta
Göttingen, 2014.

Abstract

The ability to quantify temporal information on the scale of hundreds of milliseconds is critical towards the processing of complex sensory and motor patterns. However, the nature of neural mechanisms for temporal information processing (at this scale) in the brain still remains largely unknown. Furthermore, given that biological organisms are situated in a dynamic environment, the processing of time-varying environmental stimuli is intricately related to the generation of cognitive behaviors, and as such, an important element of learning and memory. In order to model such temporal processing recurrent neural networks emerge as natural candidates due to their inherent dynamics and fading memory of advent stimuli. As such, this thesis investigates recurrent neural network (RNN) models driven by external stimuli as the basis of time perception and temporal processing in the brain. Such processing lies in the short timescale that is responsible for the generation of short-term memory-guided behaviors like complex motor pattern processing and generation, motor prediction, time-delayed responses, and goal-directed decision making. We present a novel self-adaptive RNN model and verify its ability to generate such complex temporally dependent behaviors, juxtaposing it critically with current state of the art non-adaptive or static RNN models.

Taking into consideration the brain's ability to undergo changes at structural and functional levels across a wide range of time spans, in this thesis, we make the primary hypothesis, that a combination of neuronal plasticity and homeostatic mechanisms in conjunction with the innate recurrent loops in the underlying neural circuitry gives rise to such temporally-guided actions. Furthermore, unlike most previous studies of spatio-temporal processing in the brain, here we follow a closed-loop approach. Such that, there is a tight coupling between the neural computations and the resultant behaviors, demonstrated on artificial robotic agents as the embodied self of a biological organism. In the first part of the thesis, using a RNN model of rate-coded neurons starting with random initialization of synaptic connections, we propose a learning rule based on *local active information storage* (LAIS). This is measured at each spatiotemporal location of the network, and used to adapt the individual neuronal decay rates or time constants with respect to the incoming stimuli. This allows an adaptive timescale of the network according to changes in timescales of inputs. We combine this, with a mathematically derived, generalized mutual information driven *intrinsic plasticity* mechanism that can tune the non-linearity of network neurons. This enables the network to maintain homeostasis as well as, maximize the flow of information from input stimuli to neuronal outputs. These unsupervised local adaptations are then combined with supervised synaptic plasticity in order to tune the otherwise fixed synaptic connections, in a task dependent manner. The resultant plastic network, significantly outperforms previous static models for complex temporal processing tasks in non-linear computing power, temporal memory capacity, noise robustness as well as tuning towards near-critical dynamics. These are displayed using a number of benchmark tests, delayed memory guided responses with a robotic agent in real environment and complex motor pattern generation tasks. Furthermore, we also demonstrate the ability of our adaptive network to generate clock like behaviors underlying time perception in the brain. The model output matches the linear relationship of variance and squared time interval as observed from experimental studies.

In the second part of the thesis, we first demonstrate the application of our model on behaviorally relevant motor prediction tasks with a walking robot, implementing distributed internal forward models using our adaptive network. Following this, we extend the previous supervised learning scheme, by implementing reward-based learning following the temporal-difference paradigm, in order to adapt the synaptic connections in our network. The neuronal correlates of this formulation is discussed from the point of view of the cortico-striatal circuitry, and a new combined learning rule is presented. This leads to novel results demonstrating how the striatal circuitry works in combination with the cerebellar circuitry in the brain, that lead to robust goal-directed behaviors. Thus, we demonstrate the application of our adaptive network model on the entire spectrum of temporal information processing, in the timescale of few hundred milliseconds (complex motor processing) to minutes (delayed memory and decision making). Overall, the results obtained in this thesis affirms our primary hypothesis that plasticity and adaptation in recurrent networks allow complex temporal information processing, which otherwise cannot be obtained with purely static networks. Furthermore, homeostatic plasticity and neuronal timescale adaptations could be potential mechanisms by which the brain performs such processing with remarkable ease.

CONTENTS

1 Introduction **1**
1.1 Timescales in the Brain 3
1.2 Short-term Memory Guided Behaviors 4
1.3 The Plastic Adaptive Brain 6
1.3.1 Synaptic Plasticity 6
1.3.2 Homeostatic Plasticity 9
1.4 Network Models: Temporal Information Processing with Recurrent Neural Networks 11
1.4.1 Reservoir Computing: Computing with Trajectories 13
1.5 Outline of the Thesis 16
1.6 Publications Related to the Thesis 18

2 Self-adaptive Reservoir Network for Temporal Information Processing (Methods) **19**
2.1 Computing with Input-driven Recurrent Neural Networks 19
2.1.1 Modeling Arbitrary Non-Autonomous Dynamical Systems (proof) 20
2.2 Self-adaptive Reservoir Framework 22
2.2.1 Network Model 23
2.2.2 Neuron Timescale Adaptation: Active Information Storage Rule 25
2.2.3 Homeostatic Plasticity: Information Theoretic Intrinsic Plasticity Rule . . 28
2.2.4 Synaptic Plasticity: Supervised Learning and Weight Adaptation 32
2.3 Learning with Self-Adaptive Reservoir Network 35
2.4 Summary 37

3 Robust Temporal Information Processing with Self-adaptive Reservoirs (Experiments and Results) **41**
3.1 Synthetic Time Series Processing 42
3.1.1 Benchmarking on Standard Datasets 42
3.1.2 Multiple Attractor Pattern Recognition 48
3.1.3 Near Critical Dynamics: Largest Lyapunov Exponent Analysis 52
3.1.4 Intermediate Summary 55

3.2 Timing and Delay Temporal Memory 56
3.2.1 Learning to Tell Time: Responding at Specific Intervals 56
3.2.2 Delay Temporal Memory with Artificial Agents 60
3.3 Complex Motor Pattern Processing 66
3.4 Discussion 72
3.4.1 Biological Relevance 72
3.4.2 Summary 73

4 Reservoir-based Adaptive Forward Internal Model for Complex Motor Prediction 75
4.1 Introduction 76
4.2 Neural Mechanisms for Complex Locomotion 77
4.3 Materials & Method 79
4.3.1 Reservoir-based Distributed Adaptive Forward Models 79
4.4 Results 83
4.4.1 Learning the Reservoir Forward Model (motor prediction) 83
4.4.2 Simulation Results 88
4.5 Discussion 92

5 Neuromodulated Combined Learning and Reward-based Reservoir Models 95
5.1 Introduction 96
5.1.1 Classical Conditioning in the Cerebellum 99
5.1.2 Reward learning in the Basal Ganglia 100
5.2 Material & Methods 102
5.2.1 Combinatorial Learning with Reward Modulated Heterosynaptic Plasticity 102
5.2.2 Actor-critic Reservoir Model of Basal-ganglia Learning 106
5.2.3 Input Correlation Model of Cerebellar Learning 111
5.3 Results 112
5.3.1 Robot model 113
5.3.2 Experimental setup 113
5.3.3 Cerebellar system: ICO learning setup 115
5.3.4 Basal ganglia system: Reservoir Actor-critic setup 117
5.3.5 Case I: Foraging without obstacle 119
5.3.6 Case II: Foraging with single obstacle 123
5.3.7 Case III: Dynamic foraging (reversal learning) 124
5.4 Discussion 129

6 Discussion and Outlook 133

Appendix 159
A.1 Information Theoretic Measures 159
A.2 Estimating Dynamics with Largest Lyapunov Exponent 161
A.3 Modular Neural Locomotion Control 163
A.4 Neuromodulatory combined learning (additional experimental results) 164

CHAPTER 1

INTRODUCTION

> ”How can a three-pound mass of jelly that you can hold in your palm imagine angels, contemplate the meaning of infinity, and even question its own place in the cosmos? Especially awe inspiring is the fact that any single brain, including yours, is made up of atoms that were forged in the hearts of countless, far-flung stars billions of years ago. These particles drifted for eons and light-years until gravity and change brought them together here, now. These atoms now form a conglomerate- your brain- that can not only ponder the very stars that gave it birth but can also think about its own ability to think and wonder about its own ability to wonder. With the arrival of humans, it has been said, the universe has suddenly become conscious of itself. This, truly, is the greatest mystery of all.”
>
> *Vilanayur. S. Ramachandran, The Tell-Tale Brain*

Understanding the underlying mechanisms of learning and memory emerging from a complex dynamical system like the biological brain and building intelligent systems inspired by such mechanisms, serves as one of the greatest pursuits of modern scientific research. The ability to learn and cognition are not merely the products of isolated neurons, but the properties that emerge from the underlying dynamics of a complex network of neurons in the brain. Despite considerable progress in neuroscience, computational sciences, and artificial intelligence, our understanding of such processes in the brain or emulation of biological like intelligence remains vastly constrained. The constantly changing nature of the environment we live in has resulted in exquisite evolutionary manipulation of the nervous system, leading to the ability to process and generate challenging spatial and temporal information. Imagine a scenario, where you are driving down the highway and someone tells you, ’take the *left* turn at the *next* junction’. To solve this seemingly simple statement the brain needs to perform a complex set of computations, within inherent dependence on the temporal aspects of the statement and any subsequent events. Not only you need to understand the meaning of the sentence and words such as, ’take’ and ’turn’, but also be able to hold this information temporarily till you reach the next junction and can perform the corresponding behavior of turning ’left’. Such temporary storage of available stimuli for the purpose of information processing, is referred to as working memory or temporal memory.

As such, it is obvious that timing and memory are intricately related in the brain. This is inherent in the brains ability to perform complex temporal information processing tasks like speech processing, motor processing, music perception, decision making for goal-oriented behaviors, working memory storage and processing, etc. Given that the brain is not static, but a highly adaptive[1] system, which processes can enable the initiation and execution of such temporal memory guided behaviors from neural activity ? We make the hypothesis that a combination of neural plasticity, homeostatic and adaptation mechanisms coupled with the presence of feedback loops (recurrency) in the neural circuitry give rise to such actions. Based on this hypothesis, the main focus of this thesis is to answer the question: *How can we model such adaptation for brain like temporal information processing that in turn lead to memory-guided behaviors ?*. The primary objective being not only to create a computational model of neural circuitry with inherent storage and processing of time varying stimuli, but also to use the same model to generate robust sensory-motor outputs and short-term memory guided behaviors in artificial intelligent systems.

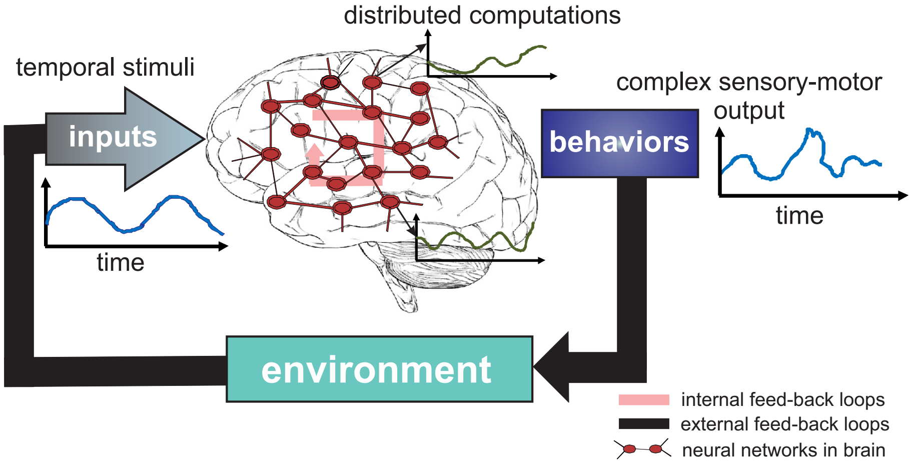


Figure 1.1: **Closed-loop approach to temporal information processing** A constant barrage of time varying stimuli perturb the resting state of the brain leading to non-trivial, non-linear, and highly distributed computations in neuronal networks in the brain. Such computations also occur over a wide distribution of timescales. With learning and adaptation, cognitive behaviors and complex sensory motor outputs, requiring robust processing of the temporal information, can be obtained. Such behaviors typically lead to changes in the environmental conditions, which in turn change the incoming stimuli to the brain networks, thus closing the input-output loop.

Given that, biological organisms as well as any artificial agents[2] are not isolated entities, but reside in an external (outside the agent) environment; their behaviors lead to changes in environ-

---

[1] Adaptive here refers to the brains ability to change at a structural or functional level across a range of time spans.

[2] Agent here refers to any artificial system, like a robot akin to some living being.

mental conditions, which in turn leads to changes in the temporal stream of sensory information that the brain receives. As such it is imperative that while modeling such information processing, we consider a closed loop approach (see Fig. 1.1). Therefore, in this thesis, unlike most modeling studies of spatio-temporal processing in the brain, we consider closed loop systems with a tight coupling between brain-like network level computations and the relevant behaviors that can be generated by such computations. We pragmatically demonstrate that by the consideration of novel adaptive and plastic mechanisms, in recurrent networks (abstraction of cortical networks) it is indeed possible to perform complex temporal information processing, that considerably outperforms non-plastic networks. Furthermore, the same principles lead to robust temporal memory guided behaviors (like motor pattern predictions and generation, goal-directed decision making, delayed responses etc.). As such, this thesis makes novel contributions to the intersections of all three fields of neuroscience, computational sciences (machine learning) and artificial intelligence (robotics).

In the following sections, we will now introduce in greater detail as well as provide the necessary background to the various aspects of brain like temporal information processing and the considerations made in this thesis towards it. Finally in the last section we provide an outline with brief overview of the various chapters in the thesis.

1.1 Timescales in the Brain

In nature, animals are capable of efficiently encoding space and time required for the learning and structuring of motor and cognitive actions. Specifically the mammalian brain processes temporal information over time scales spanning 10 orders of magnitude: from the few microseconds used for sound localization, to daily, monthly and yearly rhythms to sleep-wake, menstrual and seasonal cycles (Buonomano et al., 2009). In between, on the scale of milliseconds to few minutes, complex forms of sensory-motor processing leading to behaviors like speech recognition, motor coordination, motor prediction, and decision making for goal-directed learning, takes place (Ivry and Spencer, 2004),(Mauk and Buonomano, 2004),(Buhusi and Meck, 2005), (Buonomano, 2007). As such, we focus on this timescale of information processing and behaviors. Within this timescale, a number of different brain areas have been implicated as the key machinery behind the neural representation of time (Maniadakis et al., 2014). Among these, some of the most relevant are cerebellar event timing (Ivry and Spencer, 2004); generalized magnitude processing for time, space, and number in the right posterior parietal cortex (Bueti and Walsh, 2009), (Oliveri et al., 2008); time integration for working memory in right prefrontal cortex (Lewis and Miall, 2006),(Smith et al., 2003); coincidence detection in the fronto-striatal circuitry (Hinton and Meck, 2004) and time cells in the hippocampus computing the relation of time and distance (Kraus et al., 2013).

Such a wide spread participation of different brain regions for temporal information processing clearly advocates the key role of temporal perception in the brain as well as the intricate relationship of the different timescales that constitute the various cognitive aspects like decision making, planing, action selection, memory and recall (Rao et al., 2001),(Taatgen et al., 2007).

On a functional level, it is known that neuronal systems can adapt to the statistics of the environment over these wide range of timescales (learning, memory and plasticity) (Tetzlaff et al., 2012b), but the mechanisms for doing so are still largely unknown. Therefore, there seems to be an essential relationship between processing of temporal information and how the brain deals with the various timescales and generate relevant behaviors. In Fig. 1.2 we provide a succinct, schematic overview of the different timescales of temporal perception in the brain and their relationship to observed physiological processes, memory, behaviors and learning paradigms. In this thesis we will primarily focus on the timescale of milliseconds to a few minutes and the behaviors, memory and processes corresponding to this scale.

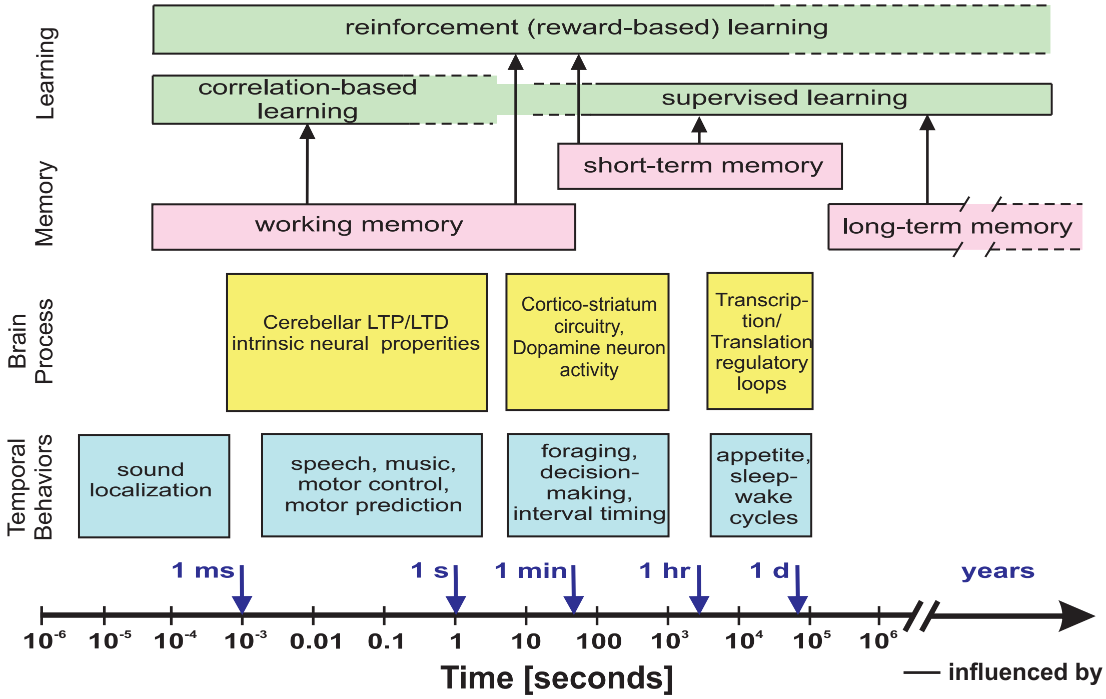

Learning
reinforcement (reward-based) learning
correlation-based learning
supervised learning
Memory
short-term memory
working memory
long-term memory
Brain Process
Cerebellar LTP/LTD intrinsic neural properities
Cortico-striatum circuitry, Dopamine neuron activity
Transcrip-tion/ Translation regulatory loops
Temporal Behaviors
sound localization
speech, music, motor control, motor prediction
foraging, decision-making, interval timing
appetite, sleep-wake cycles
1 ms
1 s
1 min
1 hr
1 d
years
10^{-6}
10^{-5}
10^{-4}
10^{-3}
0.01
0.1
1
10
100
10^{3}
10^{4}
10^{5}
10^{6}
Time [seconds]
influenced by

Figure 1.2: **Timescales in the brain and their relations to various brain processes, memory, learning and behavior** Animals can process temporal information over a wide range of timescales. Each timescale from microseconds to days accounts also for sophisticated behaviors and their inherently related memory processes and learning paradigms. Specifically in the range of few hundred milliseconds to few minutes is where the most complex temporal information processing occurs, which is needed for non-trivial sensory-motor processing, prediction, planning, as well as decision making purposes. Modified and extended from (Tetzlaff et al., 2012a)

1.2 Short-term Memory Guided Behaviors

Complex behaviors like memory guidance (also called delayed responses) and goal directed action planning involving temporal memory (short-term storage in the timescale of milliseconds to minutes) and learning can be observed not only in higher order mammals but also in insects. For instance, cockroaches use their cercal filiform hairs (wind sensitive hairs) to elicit so called ”wind-evoked escape behavior” (Beer and Ritzmann, 1993); i.e., they turn and then run away from a wind puff to their cerci generated by a lunging predator. This action perseveres slightly longer than the stimulus itself. Once the action has been activated, it will be performed even if the activating stimulus is removed to ensure safely escaping from the attack. Thus, this action reflects not only a reactive response but also a simple memory-guided behavior (transiently active internal drive) (Arkin, 1998). More complex examples can be found in mammals such as the one observed in the behavior of cats (McVea and Pearson, 2006). They use temporal memory of visual inputs in order to drive their front legs at the appropriate *time* to step over or around obstacles in their path at a time the obstacle is already invisible. There is also a unique form of predictive memory to guide the hind legs over obstacles that have already been stepped over by the forelegs. This can also be regarded as some form of predictive or forward modeling behavior (Kawato, 1999) which is a crucial aspect of temporal information processing. This type of processing can also be seen to occur even in invertebrates, allowing them to climb over large gaps almost twice the size of their body lengths (Blaesing and Cruse, 2004). Other sophisticated navigation and foraging studies with rodents have shown that they not only use spatial memory with reward learning, to navigate mazes and find food (Tolman, 1932),(Tolman and Honzik, 1930), (Olton and Samuelson, 1976), but they can also develop temporally structured behaviors, demonstrating some form of temporal memory to discriminate between long and short time intervals (Gouvea et al., 2014).

As depicted in Fig. 1.2 such short-term memory guided behaviors are intricately related to the brains ability to process time or time varying patterns of activity. Furthermore, in order to understand such temporal processing, it is important to put it in a closed-loop perspective (Fig. 1.1). As such, in this thesis we use network models with inherent time processing that can lead to similar temporal memory guided behaviors, as evaluated on artificial agents as abstractions of their biological counterparts (Arkin, 1998). Given that learning and memory is ultimately a consequence of a highly plastic brain (Dudai, 2004),(Martin et al., 2000), it is obvious that it should play a key role in the underlying temporal information processing. In the next section, we broadly explore the various facets of neuronal plasticity and put it in perspective of this thesis.

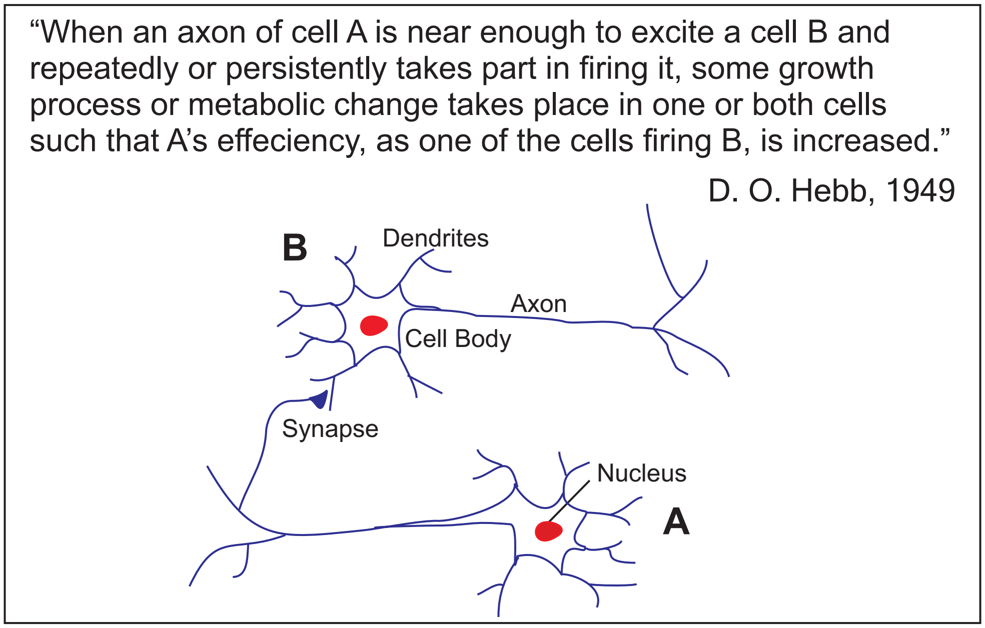
1.3 The Plastic Adaptive Brain

> ”The labor of a pianist [. . .] is inaccessible for the uneducated man as the acquisition of new skill requires many years of mental and physical practice. In order to fully understand this complex phenomenon it becomes necessary to admit, in addition to the reinforcement of pre-established organic pathways, the formation of new pathways through ramification and progressive growth of the dendritic arborization and the nervous terminals.”
>
> *Textura del Sistema Nervioso*, Santiago R. Cajal (1904)

The inherently malleable and constantly adaptive nature of the nervous system was clearly noted by Cajal (1904) when he predicted that with the acquisition of new skills the brain changes via rapid reinforcements of pre-established organic pathways, which in turn lead to the formation of new pathways (Pascual-Leone et al., 2005). Although Cajal specifically mentioned neural pathways (synapses), recent experimental and theoretical studies have confirmed that nearly every brain region demonstrates such remarkable and flexible reorganization. Widespread structural and functional alterations occur by processes of modulation of strength of synaptic connections between neurons (Abbott and Nelson, 2000), addition and deletion of connections (Holtmaat and Svoboda, 2009), changes in the intrinsic excitability of single neurons (Zhang and Linden, 2003), as well as, balancing homeostatic adaptation processes (Turrigiano and Nelson, 2004). Furthermore, the seminal studies of Merzenich and Kaas (Merzenich et al., 1983), (Merzenich et al., 1984) demonstrated that topographic reorganizations of cortical maps can be realized in an experience-dependent manner through neural plasticity, thus, highlighting the central role of brain plasticity in a lifelong learning process. Specifically, at the behavioral level, such adaptive mechanisms in the brain provides it with the crucial ability to learn and deal with environmental changes, capture and retain specific memories, process information critical for speech and motor functionality, etc. In general, neural plasticity can be divided into two broad types, namely (i) synaptic plasticity and (ii) homeostatic plasticity. As the main motivation behind this thesis is not to understand the biophysical machinery behind such plasticity mechanisms, but to use them as biological inspiration to adapt network models in order to deal with time varying stimuli and the processing of temporal information; in the next two subsections we briefly introduce the basic ideas of these two types of plasticity in the brain.

1.3.1 Synaptic Plasticity

Synaptic plasticity can be defined in simple terms as the process of strengthening or weakening of synapses connecting different neurons, facilitating the transmission of electro-chemical signals (Citri and Malenka, 2007). Specifically, it refers to the activity-dependent modification of the strength or efficacy of synaptic transmission at pre-existing synapses, caused by the changes in the amount of neurotransmitter molecules at the synapse, or by the fluctuation in conduction of post-synaptic receptors. Synaptic plasticity is thought to play key roles in the early development of neural circuitry (termed as cell assemblies) (Hebb, 1949), (Dudai, 2004) and



"When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's effeciency, as one of the cells firing B, is increased."

D. O. Hebb, 1949

Figure 1.3: **Hebb's postulate and synaptic plasticity** Schematic rendering of two biological neurons, showing a synaptic connection.

experimental evidence suggests that impairments in synaptic plasticity mechanisms contribute to several prominent neuropsychiatric disorders (Lau and Zukin, 2007). The encoding of external and internal events as complex, spatio-temporal patterns of activity within large ensembles of neurons is directly influenced by this type of plasticity of the pattern of synaptic weights that connect individual neurons comprising such ensembles or neuronal circuits. This forms the direct basis for the plasticity and memory hypothesis (Martin et al., 2000), (Martin and Morris, 2002), suggesting that activity-dependent changes in the strength of connections between neurons plays the key role towards the mechanism by which new information is stored or memory traces are encoded in the central nervous system.

The simplest theoretical foundations of such an idea was postulated by Donald Hebb, as early as 1940, where in he proposed that associative memories are formed in the brain by a process of synaptic modification that strengthens connections when presynaptic activity correlates with postsynaptic firing (Hebb, 1949) (Fig. 1.3). This has been popularly termed as Hebbian plasticity i.e. '*cells that fire together, wire together*' (Carla Shatz, 1992). The first experimental validation of Hebbian type of plasticity (showing an increase in synaptic efficacy) came from Bliss and Lomo in 1973 (Bliss and Lomo, 1973), with the report of the phenomenon of long-term potentiation (LTP). Subsequently in the year 1977, Lynch et al. (Lynch et al., 1977) found a reduction in synaptic efficacy called long-term depression (LTD). Later it was also noted that both LTP and LTD could be observed at the same synapse (Dudek and Bear, 1992). Unlike the basic Hebbian formulation of correlations between neuron firing activity, an influence of temporal signal order on plasticity was proposed by (Gerstner et al., 1996) and then experimentally validated by the

Plasticity Rule	Mathematical representation	Learning Paradigm
Generalized Hebbian rule*	$\dot{\omega}_{ij} = \eta x_i x_j$	Unsupervised Learning
Oja's rule*	$\dot{\omega}_{ij} = \eta(x_i x_j - \alpha x_i^2 \omega_{ij})$	Unsupervised Learning
BCM rule*	$\dot{\omega}_{ij} = \eta(x_i x_j (x_i - \theta_i),$ $\dot{\theta}_i = \tau_\theta (x_i^2 - \theta_i).$	Unsupervised Learning
Gradient-descent rule	$\dot{\omega}_i = \eta e x_i,$ $e = d - y$ $y = \phi(x, \omega)$	Supervised Learning
Reward-modulated Hebbian†	$\dot{\omega}_i(t) = \eta\Big(x_i(t)\xi_i(t)\big[R(t) - \langle R(t)\rangle_t\big]\Big)$	Reinforcement Learning

Table 1.1: **Simplified summary of Hebbian type rate-based synaptic plasticity rules and the related learning paradigm** Depending on the type of learning paradigm used, there can be various formulations of the basic Hebb rule based on correlations between pre- and post-synaptic neuron activity. ω_{ij} synaptic weight between neuron j and i; x_i firing rate of neuron i; t timestep; $\eta \ll 1$ learning rate ; θ_i threshold on post-synaptic activity; α positive constant; τ timescale parameter; e learning error; d supervised desired output; y Output activity of neuron; ϕ non-linear activation function; $\xi_i(t)$ exploration signal; $R(t)$ reward signal; $\langle . \rangle_t$ Mean activity in time; * the basic Hebbian plasticity rule is unstable in nature (unbounded growth due to positive correlations); * Oja's and BCM rule are stable formulations of the standard Hebbian rule. † the reward-modulated Hebbian learning rule has been adapted from (Legenstein et al., 2010) and is a generic representation for reinforcement based learning. Various modifications based on the temporal-difference error learning are also possible (Sutton, 1988),(O'Doherty et al., 2003)

findings of (Markram et al., 1997), (Magee and Johnston, 1997), (Levy and Steward, 1983), (Bi and Poo, 1998). As such this type of plasticity has been termed as spike-timing dependent plasticity (STDP).

In this thesis, we model synaptic plasticity based on the correlations between the firing rate of the pre- and post-synaptic activity of the neuron, following the spirit of the basic Hebbian conjecture without delving deep into molecular or biophysical details (Dayan and Abbott, 2003). Furthermore depending on the type of learning paradigm, specific modifications of the original Hebbian learning rule will be used (see Tab. 1.1). We will primarily consider supervised and reinforcement learning in this thesis.

1.3.2 Homeostatic Plasticity

The word homeostasis or homeostatic stems from the Greek word *homeo* meaning 'unchanging' and is a generic concept guaranteeing the ability of a system to reach the same internal state as prior to the application of an external perturbation. In neuronal systems homeostatic plasticity refers to the capacity of neurons and synapses to regulate their own excitability relative to the network activity, usually in response to an imbalance or external disturbances. It can be seen to balance the inherently unstable nature of purely Hebbian plasticity (correlations of pre- and post-synaptic activity) by modulating the activity of the synapse (Davis, 2006) or the properties of voltage-gated ion channels (Zhang and Linden, 2003). This regulates the total synaptic drive to a neuron and/or maintain the long-term average firing rate of a neuron at a critical level and therefore allows the stable operation of neuronal networks.

The two principle types of homeostatic mechanisms are namely *synaptic scaling* (SC) (Fig. 1.4 (a)) and *intrinsic plasticity* (IP)(see Fig. 1.4 (b)). SC is a mechanism that regulates the total synaptic drive received by a neuron while maintaining the relative strength of synapses established during learning (Turrigiano et al., 1998), (Turrigiano and Nelson, 2004). It has been found in several brain areas including the neocortex (Turrigiano et al., 1998), Hippocampus (Burrone et al., 2002) as well as at inhibitory synapses (Kilman et al., 2002). IP, on the other hand, is a homeostatic mechanism leading to the persistent modification of a neuron's excitability, mediated by the properties of ion channels in the neuron's membrane. It was noted that such intrinsic changes in a neuron's electrical properties, might function as part of the engram itself, or as a related phenomenon such as a trigger for the consolidation or adaptive generalization of memories (Zhang and Linden, 2003). Changes in neuronal excitability via IP lead to different outputs for the same synaptic drive. Furthermore it was experimentally observed that IP tends to reduce the intrinsic excitability of a neuron during long periods of stimulation and increase excitability during activity deprivation (Desai et al., 1999), (Zhang and Linden, 2003), (van Welie et al., 2004) (Fig. 1.5 (a) and (b)).

In our network models, we primarily consider intrinsic plasticity at single neurons level and see its influence on homeostatic regulation as well as learning. Evidence that IP accompanies, and may help mediate, learning has been obtained in both invertebrates (Drosophilia - (Daoudal and Debanne, 2003), Aplysia - (Brembs et al., 2002) etc.) and mammals (Oh et al., 2003), (Saar

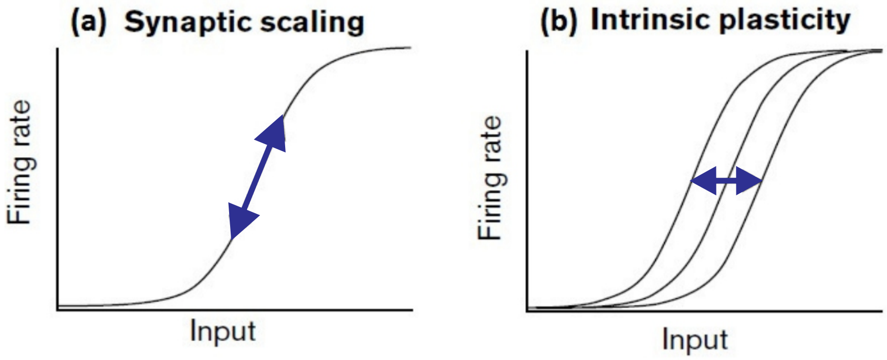


Figure 1.4: **Schematic representation of homeostatic mechanisms** (a) Synaptic scaling : By scaling the strength of all the neurons inputs up or down, the neuron's property can be shifted up or down its firing rate curve. This determines how fast the neuron fires for a given amount of synaptic drive (b) Intrinsic plasticity: The regulation of intrinsic neuronal conductance can modify the input/output curve of the neuron by shifting it left (fires more for a given synaptic drive) or right (fires less). It can also lead to modifications of the slope of the curve leading to different levels of non-linearity.

et al., 1998), (Brons and Woody, 1980), specially with associative conditioning experiments (more details in (Zhang and Linden, 2003)). Furthermore along with its role in homeostasis, IP has been implicated directly in the formation of memory engrams (Gandhi and Matzel, 2000). From an information transmission perspective (Fig. 1.5 (c) and (d)), IP can be seen to allow a neuron to exploit its full dynamic range of firing rates when coding for a given set of inputs and achieving exponential firing rate distributions in cortical neurons (Stemmler and Koch, 1999). It was also linked to information maximization and energy efficient coding at a single neuron level (Vincent et al., 2005).

In this thesis, we model IP based on the same principles of information maximization (Triesch, 2007) for a recurrent network model which has shown to induce robust homeostatic effects on network dynamics (Steil, 2007), (Schrauwen et al., 2008), as well as increased performance for information processing and memory (Verstraeten et al., 2007), (Dasgupta et al., 2012), (Dasgupta et al., 2013a). Inspired by these approaches, in this thesis we take such an information-centric view of IP, such that neurons are enabled to maximize information transfer between its input and output, as well as matching the statistics of some optimal output distribution by modulating their activation functions in an input-dependent manner (Fig. 1.5 (d)).

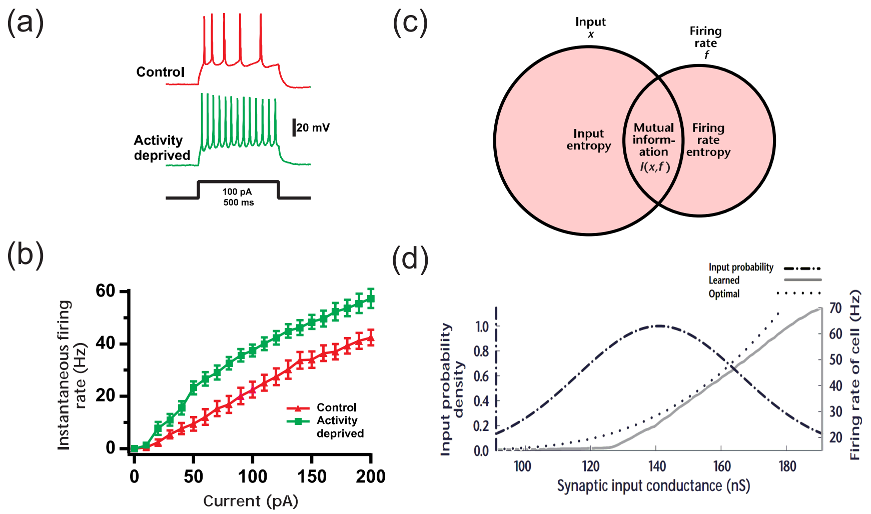


Figure 1.5: **Intrinsic plasticity** (a)-(b) chronic activity blockade resulted in an increase in the firing frequency and decrease of the spike threshold of pyramidal neurons. (a) Sample spike trains evoked by a somatic current injection in neurons grown under control and activity deprived conditions. (b) Initial instantaneous firing rate versus amplitude of current injection for control and activity-deprived neurons. Changes in the intrinsic properties of the neuron result in change in shape of the firing rate curve as a result of activity deprivation. Adapted from (Desai et al., 1999). (c) An information centric view of IP holds that the intrinsic properties of a neuron are tuned to produce the best match with whatever synaptic input it receives, i.e. to maximize the mutual information between it's input and output. This also directly relates to the idea of information maximization (d) Learning an optimal firing rate response curve assuming a mean firing rate of 30 Hz (model neuron in (Stemmler and Koch, 1999)). Given an Gaussian input distribution, IP allows neurons to adjust their firing rate responses in order to learn an optimal exponential output distribution. Adapted from (Stemmler and Koch, 1999)).

1.4 Network Models: Temporal Information Processing with Recurrent Neural Networks

In the previous section we broadly discussed plasticity in biological brains which forms the basis of learning in living organisms. However the question of how do we model such learning? still remains unclear. In order to answer this question, we take a connectionists approach. Whereby we model the actual behavioral phenomenon as the emergent process or learning outcome of the dynamics of interconnected networks of simple units (artificial neurons). This type of network models have been termed as artificial neural networks where in, the fundamental computational unit of such networks although called neurons, they only very broadly resemble their biological counterparts. Here we typically consider artificial rate-coded neurons which compute their output as a non-linear transformation (activation function) of the sum of weighted inputs (incoming synaptic connections) it receives (Fig. 1.6 (a)).

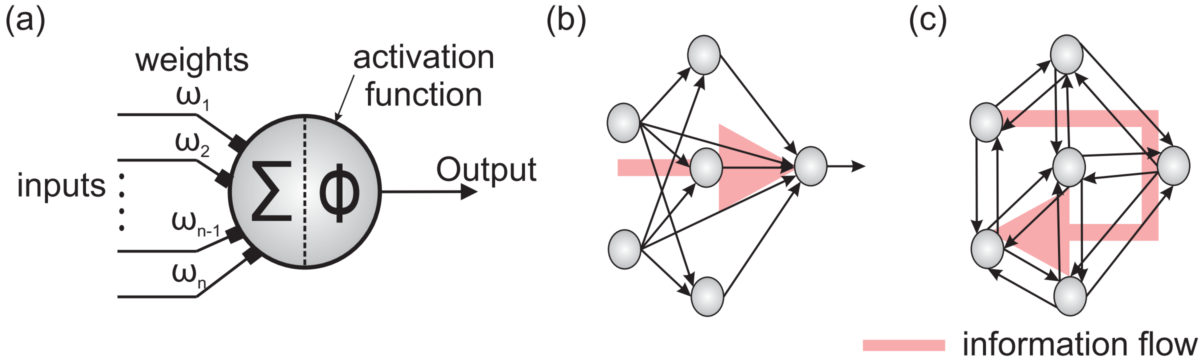


Figure 1.6: **Pictorial representation of neural network models** (a) An artificial rate coded neuron. The output is calculated as a non-linear transformation (based on activation function ϕ) of the weighted (synaptic strengths) sum of incoming inputs. (b) A typical (fully-connected) feed-forward network as a directed acyclic graph. Due to the one directional flow of information, typically there is limited fading memory of input stimuli and no internal memory of past activity (c) A fully connected recurrent neural network. Due to feed-back loops in the networks activity reverberates inside with a cyclic flow of information. This results in a broader fading memory of inputs as well as inherent memory of previous networks states.

There are two broad classes of neural networks that have been used in the past for handling time-varing input signals and solving specific temporal problems. These are namely feed-forward networks (Fig. 1.6 (b)) and recurrent networks (Fig. 1.6 (c)). Due to the lack of reverberating activity and a one directional flow of information in feed-forward networks, they have mostly been used to process non-temporal problems. Only in some cases, specific adaptations allowed feed-forward networks to incorporate in their structure an explicit representation of time (Elman and Zipser, 1988). However such explicit representation is computationally expensive as well as biologically unrealistic (Elman, 1990). Recurrent neural networks (RNN) on the other hand form the natural candidates for temporal information processing, due to their inherently dynamic nature and the existence of directed cycles inside the network, which allows reverberation of activity. As such, throughout this thesis we will concentrate on this type of neural network model. The first studies of RNNs started with the seminal works of Hopfield in 1982 and 1984 (Hopfield, 1982), (Hopfield, 1984), although Wilson and Cowan (Wilson and Cowan, 1972) originally developed the recurrent network in a biological context, a few years earlier. Using a RNN with a restricted topology of symmetric synapses, Hopfield demonstrated how to embed a large number of stable attractors into the network by setting the strengths of synapses to specific values. Trained with Hebbian plasticity this type of network could display auto-associative memory properties. However it did not consider time-varying input stimuli to drive the network, and it had very limited applicability to temporal problems. Despite the natural ability of RNNs to encode time, a universal computing ability and the subsequent development of a number of learning algorithms like Real-Time Recurrent Learning (Williams and Zipser, 1989), and Back-Propagation Through Time (Rumelhart et al., 1988), (Werbos, 1990), their usage on complex temporal problems remained restricted for a long period of time. This was largely due to the difficulty in training (Bengio et al., 1994) these networks. Furthermore, although the short-term storage of information is critical towards the ability of the brain (or a recurrent network model) to perform cognitive tasks like planning and decision making (Ganguli et al., 2008), previous models considered that the neural substrate for such memory arose from persistent patterns of neural activity, that were stabilized through reverberating positive feedback in the RNNs (Mongillo et al., 2008), (Seung, 1996) or at the single cell (Loewenstein and Sompolinsky, 2003). However, such simple attractor mechanisms are inherently incapable of remembering sequences of past temporal inputs.

1.4.1 Reservoir Computing: Computing with Trajectories

Over the last decade, an alternative idea has tried to circumvent the training problem as well as the temporal memory issue, by suggesting that an arbitrary recurrent network could store information about recent input sequences in its transient dynamics, even if the network does not formally possess information-bearing stable attractor states. This was simultaneously introduced, both from a neurobiological perspective - Liquid state machines (Maass et al., 2002) and a machine learning perspective - Echo state networks (Jaeger, 2001a), (Jaeger and Haas, 2004). In this setup, a randomly structured RNN is used as a high dimensional projection space ('reservoir') that transforms any time varying input signal into a spatial representation. Learning occurs only at the level of downstream readout networks, which can be trained to instantaneously extract relevant functions of past inputs from the reservoir, in order to guide future actions and solve spatio-temporal tasks. This type of RNN has been popularly termed as '*Reservoir Computing*' (RC) (Lukoševičius and Jaeger, 2009). The basic idea of computation in a RC is analogous to the surface of a liquid. Even though this surface has no attractors, save the trivial one in which it is flat, transient ripples on the surface can nevertheless encode information about past objects that were thrown in (Ganguli et al., 2008). This provides the inherent property of fading memory (Jaeger, 2001b), (Boyd and Chua, 1985) crucial for temporal information processing. At each time point, the reservoir network combines the incoming stimuli with a volley of recurrent signals containing a memory trace of recent inputs.

In general, for a network with N neurons, the resulting activation vector at any discrete time t, could be regarded as a point in a N-dimensional space or manifold. Over time, these points form an unique pathway (in an input or context-dependent manner) through this high-dimensional state space, also referred to as a ”neural trajectory”. The readout layer can then be trained, using supervised learning techniques, to map different parts of this state space to some desired outputs. As a result, this same concept has also been referred to as *transient dynamics* (Rabinovich et al., 2008) or *computing with trajectories* (Buonomano and Maass, 2009). This idea of computing with neural trajectories is further exciting considering that, although there is some evidence that in higher-order cortical areas simple fixed-point attractors play a part in working memory (Goldman-Rakic, 1995),(Wang, 2001), few data suggest that they contribute to the pattern recognition of complex time-varying stimuli. Thus, it is possible that in early cortical areas discrimination of temporal signals could be extracted from such high dimensional neural trajectories.

Although this type of RNN is an abstract model in general, it shares a number of essential similarities with biological neural circuits (Sussillo, 2014). A typical RC (Fig. 1.7) has the following properties:

- There are a large number of non-linear units (neurons) interconnected inside the recurrent layer.
- Strong feedback connections exist between the neurons. The non-linear activation functions, coupled with strong feedbacks and a high dimensional state space often lead to non-trivial dynamics.
- Fading memory. The system dynamics inherently contain information about the past of the input stimuli.
- The individual units works together in parallel, and in a distributed manner to implement complex computations.

Theoretically using the Stone-Weierstrass theorem (Stone, 1948), it can be proven that such liquid or reservoir computing networks can behave like universal function approximators (Maass et al., 2004), and can approximate any dynamical system under fairly mild and general assumptions (Funahashi and Nakamura, 1993). This coupled with its ability to inherently represent time (Buonomano and Maass, 2009), makes such RNNs a suitable candidate for modeling of complex spatio-temporal tasks. They can display arbitrarily complex dynamics, including regular stable dynamics (Fig. 1.7 (c)), limit cycles (Fig. 1.7 (d)), as well as chaos (Fig. 1.7 (e)). Reservoir networks have been previously successfully applied for chaotic time-series prediction and signal correction (Jaeger and Haas, 2004), (Wyffels et al., 2008), (Wyffels and Schrauwen, 2010); speech recognition (Triefenbach et al., 2010); robot learning (Hartland and Bredeche, 2007), (Kuwabara et al., 2012); epileptic seizure detection (Buteneers et al., 2009), brain-machine interface applications (Sussillo et al., 2012) etc. Despite the apparent success in machine learning applications, the application of reservoir networks to more complex temporal-processing tasks has been limited due to the large number of free parameters in the network, limited robustness to noise in reservoir activity, effect of different non-linearities activation functions on the temporal memory capacity, as well as a largely non-plastic, non-adaptive recurrent layer. Specifically, just simply creating a reservoir at random is greatly unsatisfactory.

Although it seems obvious that, when addressing specific modeling tasks, a specific reservoir design that is adapted to the task will lead to better results than a naive random creation, adaptation in RC has been a difficult problem. Most studies of adaptation in reservoir networks in order to deal with these problems has been restricted to evolutionary learning strategies (Bush and Anderson, 2005), (Jiang et al., 2008), costly gradient decent methods (Jaeger et al., 2007), specific topologies for recurrent layer (Jarvis et al., 2010), (Xue et al., 2007), or mostly by careful empirical evaluations or manual design (Lukoševičius and Jaeger, 2009). In 2009, Sussillo and Abbott (Sussillo and Abbott, 2009) introduced the 'FORCE' learning algorithm which allowed a generic reservoir network working in the chaotic domain to be trained for complex time-series modeling tasks. In further extensions, they showed that using feedback from the readout layer, it was possible to learn both recurrent as well as recurrent-to-readout weights (Sussillo and Abbott, 2012). Although this allowed for some level of plasticity in the network, no significant gain in performance was observed. More recently, Laje and Buonomano (Laje

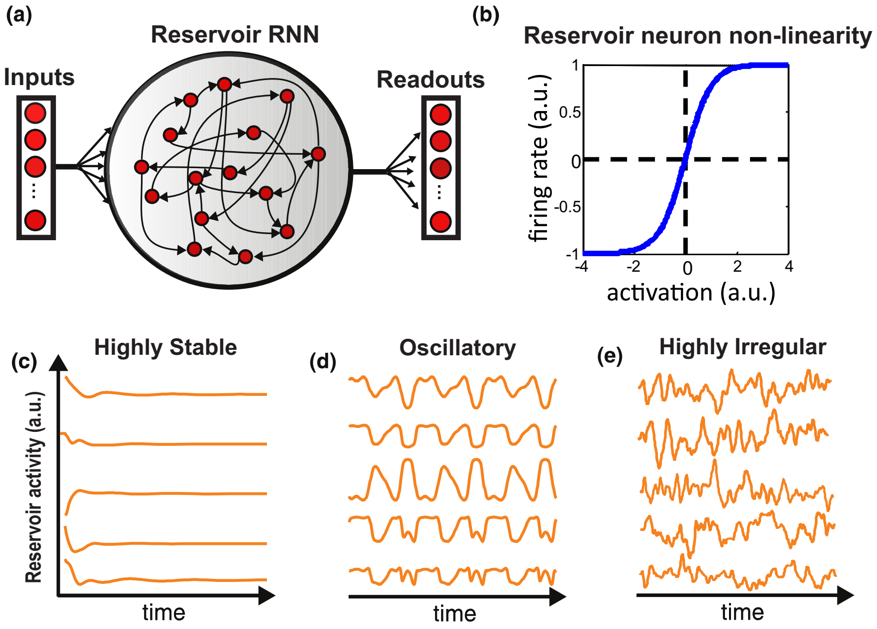


Figure 1.7: **Reservoir Computing Recurrent Neural Network** (a) A high dimensional recurrent circuit as a dynamic, distributed computing framework. Incoming time varying input stimuli project to the reservoir and influence the ongoing dynamics. The readout layer consists of neurons which compute a weighted sum of network firing rates. Synaptic connections inside the reservoir network and reservoir to readout connections can be optimized using supervised error signals. (b) Reservoir neurons typically have saturating non-linear activation functions allowing complex computation. (c) Subset of reservoir neuron activity showing stable regular dynamics (d) period oscillatory dynamics and (e) irregular chaotic dynamics of reservoir neurons. Different types of dynamics can exist inside the reservoir network, depending on the type of optimization and strength of connections. Re-plotted based on Sussillo (2014).

and Buonomano, 2013) were able to achieve coexisting stable and chaotic trajectories in a rate-based RNN model (Sussillo and Abbott, 2009) when the recurrent connections were tuned using a supervised plasticity rule, called 'innate' learning. Using the concept of dynamic attractors, they demonstrated the ability of the network to deal with perturbations or noise. However, the model still remains strictly computational with limited application to complex spatio-temporal tasks (similar to the machine learning problems tested with non-adaptive reservoirs) or generating memory-guided cognitive behaviors.

From the perspective of information processing in the brain, extension of RNNs with the principles of self-organization is crucial as it constitutes the basic computational units in the cortex (Douglas and Martin, 2004). As such it is imperative to understand the interaction of different plasticity mechanisms in the brain and how they can lead to self-organization of recurrent network models, as well as improve the performance of non-adaptive, static reservoir networks. In the computational neuroscience community, only few attempts have been made in this direction in a successful manner (Lazar et al., 2007), (Lazar et al., 2009), (Toutounji and Pipa, 2014) showing a self-organized network via the interaction of plasticity and homeostatic mechanisms.

However they have typically considered simplified binary neuron models with specific K-winner take all network topology, as well as restricted the computation of the reservoir network as linear classifiers without the requirements for cognitively relevant temporal processing. As such, there exists a large gap between the results obtained from the computational neuroscience approaches to RNN modeling as compared to the previously discussed machine learning based approaches or models. In this thesis, we primarily bridge this gap by introducing novel homeostatic mechanisms and adaptation of the RNN in an information-centric manner, which when coupled with synaptic plasticity can not only achieve a biologically plausible, temporal information processing model, but also provide superior performance in cognitively based spatio-temporal behaviors as compared to the state of the art with non-plastic networks.

1.5 Outline of the Thesis

In the previous sections we provided an overview of the main hypothesis and goal of this thesis, along with a generic review of some of the essential background of this study. We now very briefly describe the contents of each chapter. This thesis, is organized in the following manner:

1. **Chapter 2**: Introduces the input-driven recurrent neural networks (reservoir networks) as non-autonomous dynamical systems and proves that such reservoir networks can approximate finite time trajectories of any arbitrary time-invariant non-autonomous dynamical system. We then provide detailed theoretical background and mathematical description of the self-adaptive reservoir network (SARN) introduced in this thesis. We introduce novel information centric plasticity mechanisms, namely intrinsic plasticity and single neuron timescale adaptation, along with supervised synaptic plasticity of network connections. Details of the learning procedure is provided, along with a starting example of relatively complex temporal processing task (having inherently two different timescales), in order highlight the learning and adaptation mechanism in SARN as compared to previous static models. The chapter ends with a short summary.
2. **Chapter 3:** In this chapter, we provide elaborate experimental results obtained by testing SARN on various temporal information processing tasks relevant in the fast timescale of a few milliseconds to minutes. The tasks were broadly classified as synthetic time series processing (various standard benchmark tests), delay temporal memory and sequence learning with artificial agents and complex motor pattern generation. We also clearly demonstrate the ability of SARN to robustly encode both stable and chaotic attractors in the same network which was hitherto not possible in static reservoir networks. Furthermore, the effect of plasticity and adaptation on the reservoir dynamics is accessed using Lyapunov stability analysis. The chapter ends with a discussion of the results in perspective of other recent recurrent network models, as well as a brief discussion on the biological plausibility of this model.
3. **Chapter 4**: In this chapter, we introduce self-adaptive reservoir based forward internal models, that can be applied on walking robots, in order to make successful motor prediction. We clearly demonstrate that using a closed loop approach, SARN based forward

models outperform previous state of the art methods, and can generate complex locomotive behaviors. The chapter ends with a short discussion of the results.

4. **Chapter 5**: In this chapter we extend the previous supervised learning setup of SARN to a more generic reward learning scheme. Specifically we demonstrate the application of SARN as a model of the basal-ganglia brain circuitry, which in combination with a correlation learning based model of the cerebellum can lead to efficient goal-directed decision making. We also introduce a novel reward modulated heterosynaptic plasicity rule that can lead to such a combined learning. Furthermore, it is clearly demonstrated that SARN outperforms traditional feed-forward neural network models for reward learning, specially in scenarios with inherent dependence on memory of incoming stimuli. We end the chapter with a brief discussion of the results
5. **Chapter 6:** Here we discuss the main contributions of this thesis along with some relevant future outlook.

1.6 Publications Related to the Thesis

Some portions of chapter 2 and chapter 3 are based on the following papers:

> **Dasgupta, S.**, Wörgötter, F., & Manoonpong, P. (2012). Information theoretic self-organised adaptation in reservoirs for temporal memory tasks. In *Engineering Applications of Neural Networks*, (pp. 31-40, 311), doi: 10.1007/978-3-642-32909-8_4. Springer Berlin Heidelberg.
>
> **Dasgupta, S.**, Wörgötter, F., & Manoonpong, P. (2013). Information dynamics based self-adaptive reservoir for delay temporal memory tasks. *Evolving Systems*, 4(4), 235-249, doi: 10.1007/s12530-013-9080-y.
>
> **Dasgupta, S.**, Manoonpong, P., & Wörgötter, F. (2014). Reservoir of neurons with adaptive time constants: a hybrid model for robust motor-sensory temporal processing. (*in preparation*).

Large portion of chapter 4 is based on:

> *Manoonpong, P., ***Dasgupta, S.**, Goldschimdt, D., & Wörgötter, F. (2014). Reservoir-based online adaptive forward models with neural control for complex locomotion in a hexapod robot. *Neural Networks (IJCNN), 2014 International Joint Conference on*, (pp.3295,3302), 6-11 July 2014, doi: 10.1109/IJCNN.2014.6889405. *equal contribution
>
> **Dasgupta, S.**, Wörgötter, F., & Manoonpong, P. (2014). Distributed reservoir forward models with neural control enable complex locomotion in challenging environments. *Frontiers in Neurorobotics*, (*submitted*).

Finally, Large portions of chapter 5 is based on the following two papers:

> **Dasgupta, S.**, Wörgötter, F., Morimoto, J., & Manoonpong, P. (2013). Neural combinatorial learning of goal-directed behavior with reservoir critic and reward modulated hebbian plasticity. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on* (pp. 993-1000), doi: 10.1109/SMC.2013.174. IEEE.
>
> **Dasgupta, S.**, Wörgötter, F., & Manoonpong, P. (2014). Neuromodulatory adaptive combination of correlation-based learning in cerebellum and reward-based learning in basal ganglia for goal-directed behavior control. *Frontiers in Neural Circuits*, 8:126, doi: 10.3389/fncir.2014.00126.

CHAPTER 2

Self-adaptive Reservoir Network for Temporal Information Processing (Methods)

> ”Henceforth *space* by itself, and *time* by itself, are doomed to fade away into mere shadows, and only a kind of union of the two will preserve an independent reality.”
>
> —*Hermann Minkowski (1906)*

In this chapter the theoretical background and detailed description of our novel plastic reservoir based recurrent neural network model, called self-adaptive reservoir network (SARN) is presented. In the first section we motivate the idea of computation with such networks from a dynamical systems point of view and show that reservoir type RNN can model any non-autonomous dynamical system to arbitrary degree of accuracy. We next introduce the SARN architecture and present description of network dynamics. This is followed by the three levels of plasticity and unsupervised adaptation introduced in this thesis, namely (i) individual neuron time constant adaptation, (ii) intrinsic plasticity and (iii) supervised synaptic plasticity of network connection weights. Finally, we demonstrate the learning mechanism and also evaluate the performance of our adaptive reservoir network compared to static reservoirs by using an artificial time series modeling task.

2.1 Computing with Input-driven Recurrent Neural Networks

The brain is a complex dynamical system with underlying temporally intricate dynamics which is greatly difficult to unravel and comprehend (Siegelmann, 2010). Due to the dynamical properties of brain activity, recurrent neural networks (RNN) has been a natural choice to model systems with brain like characteristics or understand the underlying principles of learning and memory. As a result of the internal feedback loops, RNNs are natural dynamical systems, where the network state is dependent on its own previous history. Traditionally they have been modeled as autonomous dynamical systems, which does not receive any external influence or has a time invariant unchanging evolution in the presence of a constant input. However, much like the brain, most natural systems are subject to time-dependent variations in their own features, as well as being perturbed by temporally varying external forces (Bressler and Kelso, 2001), (Rabinovich et al., 2008). Mathematically these systems fall under the category of non-autonomous dynamical systems or input-driven (time variant) systems (Manjunath and Jaeger, 2013), (Kloeden and Rasmussen, 2011). Therefore, to model brain like temporal information processing or other naturally occurring non-autonomous systems, it is important to consider the influence of time-varying inputs on RNNs. As such, here, we consider such input driven RNNs inspired by the reservoir computing framework.

Before we introduce our reservoir based RNN, it is pertinent to answer the question, *can such a RNN indeed approximate arbitrary non-autonomous dynamical systems?*. Previous work on both discrete-time (Jin et al., 1995) and continuous time (Funahashi and Nakamura, 1993) RNNs proved that they can behave as universal approximator for autonomous systems upto an arbitrary degree of accuracy. However, they focused only on time invariant systems. Based on the work of Nakamura and Nakagawa (2009) and Chow and Li (2000), here we show that a generic class of input-driven RNN can also model the finite time trajectory of any time-variant non-autonomous system. Moreover, reservoir networks form a special case of such input-driven RNNs.

2.1.1 Modeling Arbitrary Non-Autonomous Dynamical Systems (proof)

In the following proof, we will show that a generic class of input-driven RNN, can model any arbitrary non-autonomous (time variant) dynamical system upto some finite time trajectory, and by corollary can also model any time variant system, without external inputs. Furthermore, the specific model (reservoir networks) we consider in this thesis, form a special case of the general input-driven RNN, and thus poses great computational capability for time-varying external stimuli as well as dependence of their own time dependent properties, similar to biological brains.

An input-driven RNN can be generally expressed in the following form:

$$\dot{\mathbf{x}}(t) = -\frac{\mathbf{x}(t)}{\tau} + \mathbf{f}(\mathbf{W}_1, \mathbf{x}(t), \mathbf{W}_2, \mathbf{u}(t)). \tag{2.1}$$

where, $\mathbf{x} \in \mathbb{R}^N$ and $u \in \mathbb{R}^K$ are the neural state and inputs vectors, $\mathbf{W}_1 \in \mathbb{R}^{N \times N}$, $\mathbf{W}_2 \in \mathbb{R}^{K \times N}$ are the recurrent and input weight matrices, respectively. τ is the individual neuron time constant. For simplicity, here we consider τ to be fixed. $\mathbf{f} : \mathbb{R}^N \times \mathbb{R}^K$ is a bound, smooth, and increasing function of 1-Lipschitz type[1]. Typically, $\mathbf{f}(.) = \tanh(.)$, such that: $\mathbf{f}(0) = 0$, $\mathbf{f}'(0) = 1$ and $\mathbf{f}'(z) > 0$, $z\mathbf{f}''(z) \leq 0$.

---

[1] A function $f(x)$ satisfies the Lipschitz condition of order k at $x = 0$, if $|f(h) - f(0)| \leq C|h|^k$ for all $|h| < \epsilon$, where C and k are independent of h, $C > 0$ and there is a fixed upper bound for all k for which a finite C exists.

Let $\mathbf{y} = (y_1, y_2, ..., y_n)^T$ be a point in an n-dimensional Euclidean space $\mathbb{R}^n$. Then from Chow and Li (2000) and Nakamura and Nakagawa (2009) we have the following *lemma*:

lemma 1: Let $X \subset \mathbb{R}^n$ and $U \subset \mathbb{R}^K$ be open sets, $L_X \subset X$ and $L_U \subset U$ be compact sets and $\mathbf{f} : X \times U \to \mathbb{R}^n$ be a continuous mapping of 1-Lipschitz type. Given a time invariant autonomous dynamical system of the form

$$\dot{\mathbf{y}}(t) = \mathbf{f}(\mathbf{y}(t), \mathbf{u}), \qquad \mathbf{y}(t) \in X, \mathbf{u} \in U, t \in [0, T] \in \mathbb{R} \tag{2.2}$$

with an initial state $\mathbf{y}(0) \in L_X$, then, for an arbitrary $\epsilon > 0$, $\exists N \in \mathbb{Z}$ and an RNN of type Eq. 2.1, which has an appropriate initial state $\mathbf{x}(0)$ and a small enough $\tau > 0$ such that for any input $\mathbf{u} : [0, +\infty) \to L_U$. Then the following holds:

$$\max_{t \in [0,T]} \|\mathbf{y} - \mathbf{x}\| < \epsilon, \qquad 0 < T < \infty \tag{2.3}$$

where $\mathbf{x} \in \mathbb{R}^N$ is the internal neural state of the RNN from which outputs are obtained.

Hence, this lemma shows that a generic input-driven RNN of type Eq. 2.14 can model or approximate the finite time $(0 < T)$ trajectory of any time invariant autonomous system. We can easily extend this lemma to obtain the following theorem, for time-variant non-autonomous dynamical systems.

Theorem 1: Let $X \subset \mathbb{R}^n$ and $U \subset \mathbb{R}^K$ be open sets, $L_X \subset X$ and $L_U \subset U$ be compact sets and $\mathbf{f} : X \times U \times \mathbb{R} \to \mathbb{R}^n$ be a continuous mapping of 1-Lipschitz type. Given a time variant non-autonomous dynamical system of the form,

$$\dot{\mathbf{y}}(t) = \mathbf{f}(\mathbf{y}(t), \mathbf{u}(t), t), \qquad \mathbf{y}(t) \in X, \mathbf{u}(t) \in U, t \in [0 < T] \in \mathbb{R} \tag{2.4}$$

with an initial state $\mathbf{y}(0) \in L_X$. Then, for an arbitrary $\epsilon > 0$, $\exists N \in \mathbb{Z}$ and an RNN of type Eq. 2.1, which has an appropriate initial state $\mathbf{x}(0)$ and a small enough $\tau > 0$ such that for any input $\mathbf{u} : [0, +\infty) \to L_U$. Then the following holds:

$$\max_{t \in [0,T]} \|\mathbf{y} - \mathbf{z}\| < \epsilon, \qquad 0 < T < \infty \tag{2.5}$$

where $\mathbf{z} \in \mathbb{R}^n$ are the internal neural state of the RNN from which outputs are obtained, and $\mathbf{x} \in \mathbb{R}^N$ are all the remaining neural states of the network.

Proof: Consider, $\check{\mathbf{y}}(t) = \begin{pmatrix} \mathbf{y}(t) \\ t \end{pmatrix} \in \mathbb{R}^{n+1}$, where $\check{\mathbf{y}}_{n+1}(t) = t$. Therefore, we can extend the n-dimensional vector $\mathbf{y}(t)$ to a $(n+1)$-dimensional vector in $\check{\mathbf{y}}(t)$.

As a result, we can reformulate Eq. 2.4 into an equivalent time-invariant form similar to Eq. 2.2:

$$\dot{\check{\mathbf{y}}}(t) = \check{\mathbf{f}}(\check{\mathbf{y}}(t), \mathbf{u}(t)), \qquad \check{\mathbf{y}}(t) \in (X \times \mathbb{R}), \mathbf{u}(t) \in U, \tag{2.6}$$

Here, $\check{\mathbf{f}} : (S \times \mathbb{R}) \times U \rightarrow \mathbb{R}^{n+1}$ is also a 1-Lipschitz type continuous mapping. The initial state is $\check{\mathbf{y}}(0) = \begin{pmatrix} \mathbf{y}(0) \\ 0 \end{pmatrix}$. Therefore, when $\mathbf{y}(0) \in L_X$, $\check{\mathbf{y}}(0) \in L_X \times [0, T]$. This is a compact subset of $X \times \mathbb{R}$. Hence, the reformulation in Eq. 2.6 satifies Lemma 1.

Hence we can say that for an arbitrary $\epsilon > 0$, $\exists N \in \mathbb{Z}$ and an RNN of type Eq. 2.1, which has an appropriate initial state $\mathbf{x}(0)$ and a small enough $\tau > 0$ such that for any input $\mathbf{u} : [0, +\infty) \rightarrow L_U$, the following holds:

$$\max_{t \in [0,T]} \|\check{\mathbf{y}}(t) - \check{\mathbf{z}}(t)\| < \epsilon, \qquad 0 < T < \infty \tag{2.7}$$

Where $\check{\mathbf{z}} \in \mathbb{R}^{n+1}$ are the neural states of $(n+1)$ units in the network from which outputs are achieved and $\mathbf{x} \in \mathbb{R}^{N-1}$. Let $\check{\mathbf{z}} = (\mathbf{z}, \check{\mathbf{z}}_{n+1})^T$, then $\mathbf{z} \in \mathbb{R}^n$ is the neural states of the first n units of the RNN from which outputs are drawn. Hence, from the definition of Euclidean norm $\|.\|$ we have

$$\|\mathbf{y}(t) - \mathbf{z}(t)\|^2 + (\check{\mathbf{y}}(t) - \check{\mathbf{z}_{n+1}})^2 = \|\check{\mathbf{y}}(t) - \check{\mathbf{z}}(t)\|^2. \tag{2.8}$$

This implies,

$$\max_{t \in [0,T]} \|\mathbf{y}(t) - \mathbf{z}(t)\| < \max_{t \in [0,T]} \|\check{\mathbf{y}}(t) - \check{\mathbf{z}}(t)\| < \epsilon, \qquad 0 < T < \infty \tag{2.9}$$

Hence, Theorem 1 is proved.

This shows that the RNN of type Eq. 2.1 with a size of N can approximate the finite time trajectory of a time variant non-autonomous dynamical system, where the RNN internally uses n units to provide outputs.

Input-driven RNN of the reservoir computing type (Sussillo, 2014),(Maass et al., 2002) as presented in this work, are a special case of the generic RNN in Eq. 2.1 such that,

$$\mathbf{f}(\mathbf{W}_1, \mathbf{x}(t), \mathbf{W}_2, u(t)) = \mathbf{W}_1 \sigma(\mathbf{x}(t)) + \mathbf{W}_2 \mathbf{u}(t). \tag{2.10}$$

Here the function $\sigma(.)$ is also a 1-Lipschitz type (typically tanh or sigmoid) having the same properties as $\mathbf{f}(.)$. As such, this result shows that such input-driven RNNs form a powerful system that can be used to generate complex time-varying patterns of activity. Considering, the brain is a complex dynamical system, which in turn is stimulated by a multitude of temporal signals, these RNN models can be used as an abstraction of the brains ability to compute with such changing stimuli for robust temporal information processing.

2.2 Self-adaptive Reservoir Framework

In this section we formally introduce the self-adaptive recurrent neural network framework. We start with the description of the recurrent network model followed by detailed explanation of the homeostatic plasticity and adaptation mechanisms introduced in this thesis, namely (i) neuron timescale adaptation based on active information storage measure, and (ii) the self-organized adaptation of reservoir neurons inspired by intrinsic plasticity and (iii) this is followed by overview of the learning objective and training procedure for supervised synaptic plasticity in the network, in order to perform different temporal information processing tasks from within a supervised learning setup.

2.2.1 Network Model

The self-adaptive RNN model based on the reservoir computing framework is depicted in Fig. 2.1. The basic setup can be divided into three layers: input, hidden or internal, and readout layers. Internal layer consists of a large recurrent neural network driven by time-varying stimuli. These driving signals are provided by the input layer. Due to the dynamic reservoir, the network exhibits a wide repertoire of nonlinear activity. This is then combined into desired output signals at the readout layer, using a suitable supervised training of the reservoir neuron to read-output connectivity. The RNN dynamics can be formally defined by the following equations:

$$\tau_i \dot{x}_i(t) = -x_i(t) + g \sum_{j=1}^{N} W_{ij}^{rec} r_j(t) + \sum_{j=1}^{K} W_{ij}^{in} u_j(t) + W_i^{fb} z(t) + B_i, \tag{2.11}$$

$$r_i(t) = \mathtt{tanh}(a_i x_i(t) + b_i), \tag{2.12}$$

$$z(t) = [\mathbf{W}^{out}]^T \mathbf{r}(t). \tag{2.13}$$

The RNN model consists of N neurons, such that the membrane potential at the soma (at time t) of the reservoir neurons, resulting from the incoming excitatory and inhibitory synaptic inputs, is given by a N dimensional vector of neuron state activation's, $\mathbf{x}(t) = x_1(t), x_2(t), ..., x_N(t)$. Here the RNN does not explicitly model action potentials, but describes neuronal firing rates, where in, the continuous variable $r_i(t)$ is the instantaneous firing rate (N dimensional) of the reservoir neurons and is calculated as a non-linear saturating function of the state activation $x_i(t)$ (Eq. 2.12). The parameters a_i and b_i govern the slope of the firing rate curve and act as a bias signal to the reservoir neurons, respectively. Tuning these parameters allows the non-linearity to be shaped in terms of the input distribution. This mechanism forms the essence of the intrinsic plasticity scheme explained in section 2.2.3. Each reservoir neuron i, receives inputs from other neurons in the network with firing rates $r_j(t)$ via synaptic connections of strength W_{ij}^{rec} along with incoming stimuli from the K dimensional input $u_k(t)$ via synapses of strength W_{ij}^{in}. Each reservoir neuron also receives an auxiliary bias signal B_i. The parameter g (Sompolinsky et al., 1988),(van Vreeswijk and Sompolinsky, 1996) acts as the scaling factor for the recurrent connection weights allowing different dynamic regimes from stable ($g < 1$) to

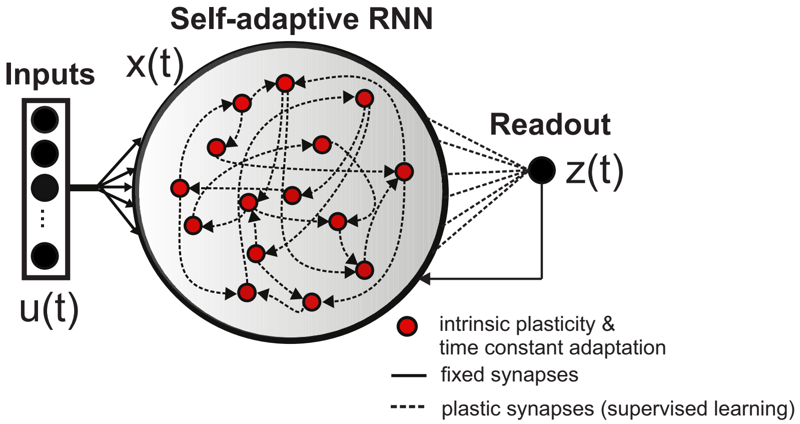

Self-adaptive RNN
Inputs
x(t)
Readout
z(t)
u(t)
intrinsic plasticity & time constant adaptation
fixed synapses
plastic synapses (supervised learning)

Figure 2.1: **The self-adaptive reservoir network architecture** The basic setup can be divided into three layers. The first payer consists of input neurons that project time varying signals $\mathbf{u}(t)$ with randomly assigned synaptic weights $\mathbf{W}^{in}$ to the recurrent layer. The recurrent layer consists of initially randomly connected neurons with synaptic weights $\mathbf{W}^{rec}$. Each recurrent layer neuron (also referred as reservoir neuron) undergoes intrinsic plasticity to adjust its non-linearity based on the inputs, as well as autonomous adaptations of their individual time constants. $\mathbf{x}(t)$ is the vector of all recurrent neuron states. Output from the recurrent layer projects on to readout neurons (for simplicity only one readout neuron is considered here) with synaptic weights $\mathbf{W}^{out}$. Readout neuron firing rate is denoted by $z(t)$. Feedback connections ($\mathbf{W}^{fb}$) if present are also randomly selected. Only the $\mathbf{W}^{out}$ and $\mathbf{W}^{rec}$ connections are plastic and learned by supervised training.

highly irregular chaotic ($g > 1$), being present in the reservoir. Similar to the recent works from (Sussillo and Abbott, 2009), the network is initialized with g such that the network exhibits chaotic dynamics as spontaneous behavior before learning and maintains stable dynamics after learning, with the help of plasticity and adaptation (sections 2.2.3 and 2.2.2). The neuronal time constant is given by the parameter τ_i defined for each reservoir neuron, and helps to determine the timescale of local neural dynamics. Although most previous models (Sussillo and Abbott, 2009),(Laje and Buonomano, 2013) have considered a fixed global time constant, in order to adapt to the temporal structure (timescales) of incoming inputs local adaptation of neuronal time constants forms a crucial link (Mozer, 1993), (Pearlmutter, 1995). As such we adapt this parameter according to a novel local information dynamics rule (section 2.2.2). Based on finite difference approximation of the RNN dynamics with a suitable time increment Δt (see Eq. 2.14), the ratio of $\Delta t/\tau_i$ controls the speed of single neuron dynamics (can be imagined as neuron leak term) and will be a value in $[0, 1]$. The output from the network is provided at the read-out layer, in terms of a linear output of the network state z(t) [2]. Although typically there can be multiple output neurons connected to the recurrent layer, here we depict a single neuron for simplicity. The output neuron receives inputs from the reservoir via the synaptic connections of strength $\mathbf{W}^{out}$ and sends inputs back into the reservoir via synapses with strengths W_i^{fb}. In general, initially the input weights $\mathbf{W}^{in}$, recurrent weights $\mathbf{W}^{rec}$ and feedback weights $\mathbf{W}^{fb}$ are

---

[2] Depending on the learning task, it is also possible to use a non-linear saturating function like *tanh* to transform the output signal, similar to Eq. 2.12

chosen randomly. Unless otherwise stated, $\mathbf{W}^{in}$ and $\mathbf{W}^{fb}$ are drawn from a uniform distribution $[-1, 1]$ while the recurrent weights $\mathbf{W}^{rec}$ are drawn from a normal distribution with zero mean and standard deviation (s.d.) $g^2/\sqrt{p_c N}$. Here p_c controls the probability of connections inside the reservoir recurrent layer and is typically set between 10% to 50%.

In all computer simulations and experiments, the actual updating of $x_i(t)$ is calculated using finite difference approximation of Eq. 2.11.

$$x_i(t+\Delta t) = \left(1 - \frac{\Delta t}{\tau_i}\right) x_i(t) + \frac{\Delta t}{\tau_i}\left(g \sum_{j=1}^{N} W_{ij}^{rec} r_j(t) + \sum_{j=1}^{K} W_{ij}^{in} u_j(t) + W_i^{fb} z(t) + B_i\right), \quad (2.14)$$

where Δt is the increment of time.

2.2.2 Neuron Timescale Adaptation: Active Information Storage Rule

In our model of the reservoir RNN (Eq. 2.14), every neurons membrane potential ($x_i(t)$) is influenced not only by current synaptic inputs, but also by their previous state. As such, here, the decay rate of each reservoir neuron's membrane potential is governed by the local neuronal time constant τ_i, analogous to the leak current of membrane potential in real neurons (Koch et al., 1996). One might consider this decay rate to correspond to an integrating time window of the neuron, in the sense that the decay rate indicates the degree to which the earlier history of synaptic inputs affects the current state. When the τ_i value of a neuron is large, the activation of the neuron changes slowly, because the internal state potential is strongly affected by the history of the neurons potential. On the other hand, when the τ_i value of a neuron is small, the effect of the history of the unit's potential is also small, and thus it is possible for activation of the neuron to change quickly. In other words, τ_i corresponding to each neuron i in the reservoir, acts as a local memory term (Yamashita and Tani, 2008). Therefore, autonomous input dependent adaptation of this quantity, can allow the neuron to robustly adjust its dynamics to slow or fast (timescale) changing temporal patterns in the input stimuli.

In order to account for an adjustable neuronal decay rate (time constant) as a model of membrane leak current and local neuron memory, it is important to be able to quantify the dynamics of distributed computation within the reservoir. However, given the complexity and non-autonomous nature of such large recurrent networks, using traditional non-linear dynamics approaches for this purpose is highly restrictive (Manjunath and Jaeger, 2013). As such we use a novel information theoretic measure (see information theoretic preliminaries in appendix A.1) called input driven active information storage, which allows us to quantify the local information dynamics of storage inside such complex networks (Wibral et al., 2014b).

Active information storage (AIS) was originally introduced by (Lizier et al., 2012) in the context and cellular automatas, and then subsequently extended to the framework of reservoir computing (Dasgupta et al., 2012), (Dasgupta et al., 2013a). It is in principle based on the idea of information storage, which can be defined as *the information in an agent, process or variable's past that can be used to predict its future* (Wibral et al., 2014a). Existing information theoretic quantities like, *excess entropy* (Crutchfield and Feldman, 2003) or *predictive information* (Bialek et al., 2001) and *statistical complexity* (Crutchfield and Young, 1989), provides a measure of this stored information. However it captures the total storage used or relevant in the future of the process or agent. Since, we are dealing with neuronal networks driven by time-varying inputs, the arbitrary future states of a given neuron is unknown and hence, the total storage that is *currently in use* is more relevant. AIS helps to quantify precisely this information at each point in time.

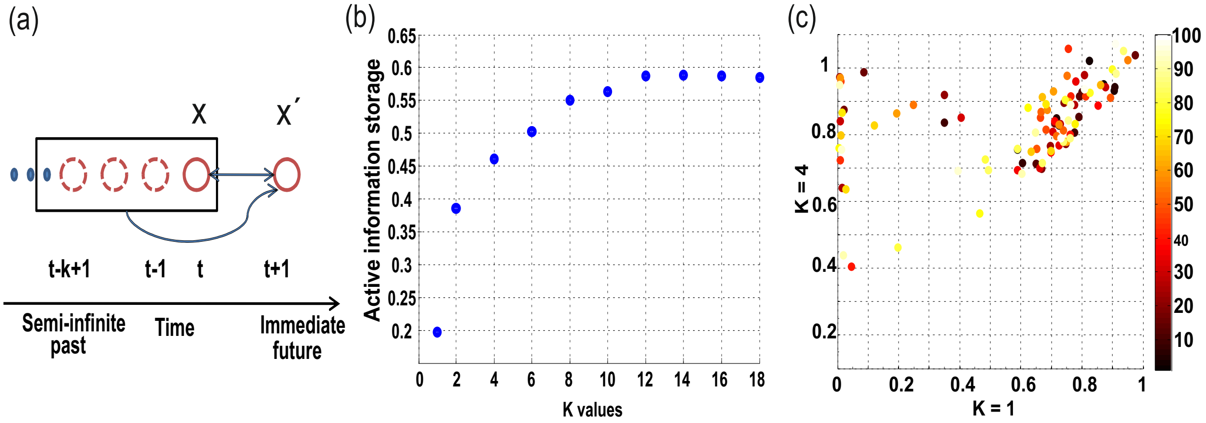


Figure 2.2: **Active information storage in reservoir neurons** (a) Pictorial representation of active information storage (AIS) calculated for a single neuron state x and its immediate future state x' (solid circle: present and next time step states of the neuron; dotted circle: previous states of the same neuron). (b) Active information storage convergence: plot of estimated AIS versus the history length k. (c) Plot of the change in local active information storage values (unaveraged) for 100 neurons with baseline history length $k = 1$ versus $k = 4$. Typically as k increases there is a change in the local estimations of AIS with some neurons showing much higher values (Colormap represents the different neurons (1-100)).

Formally, AIS A_x is the average mutual information ($\mathbf{I}$ see information theoretic preliminaries in appendix A.1) between the semi-infinite past of the network state $x^{(k)}$ and its immediate future state x' (see Fig. 2.2 (a)), rather than the whole future:

$$A_x = \lim_{k \to \infty} \mathbf{I}(x^{(k)}; x'). \tag{2.15}$$

Unfolded in time, the instantaneous AIS for a variable x is the local (or un-averaged) mutual information between its semi-infinite past $x_t^{(k)} = \{x_{t-k+1}, ..., x_{t-1}, x_t\}$ and its next state x_{t+1} at the time step $t + 1$ calculated for finite-k estimations. Hence, the local information storage is defined for every spatio-temporal point within the recurrent network. The local unaveraged information storage can take both positive as well as negative values, while the active (average) information storage $A_x(i, k) = \langle a_x(i, t, k) \rangle_t$ is always positive and bounded by the average information capacity of a single neuron state. The local information storage for a reservoir neuron state x_i is given by[3]:

$$a_x(i, t+1) = \lim_{k \to \infty} \mathbf{I}(x_{i,t}^{(k)}; x_{i,t+1}) = \lim_{k \to \infty} \log \left(\frac{P(x_{i,t}^{(k)}, x_{i,t+1})}{P(x_{i,t}^{(k)})P(x_{i,t+1})} \right). \tag{2.16}$$

Recall that, biological networks as our model of the reservoir RNN are non-autonomous dynamical systems which are driven by time varying inputs. Therefore if inputs are changing over time, the local dynamics of the states of the reservoir neurons need to account for this in quantifying information storage. As such the previous formulation in Eq. 2.16 is incomplete in the context of input driven neural systems. Therefore, in order to correctly estimate AIS, one needs to condition out the current input into the network (u_{t+1}):

$$\begin{aligned} a_x(i, t+1) = \lim_{k \to \infty} \mathbf{I}(x_{i,t}^{(k)}; x_{i,t+1} | u_{t+1}), \\ = \lim_{k \to \infty} \log \left(\frac{P(x_{i,t}^{(k)}, x_{i,t+1} | u_{t+1})}{P(x_{i,t}^{(k)})P(x_{i,t+1} | u_{t+1})} \right), \\ = \lim_{k \to \infty} \log \left(\frac{p(x_{i,t+1} | x_{i,t}^{(k)}, u_{t+1}}{p(x_{i,t+1} | u_{t+1})} \right). \end{aligned} \tag{2.17}$$

where $a_x(i, t+1, k)$ represents finite-k estimates. Using a history length of $k = 1$ is the natural starting choice for calculations of the estimates, however with increasing values of $k \to \infty$, the estimates tend towards the actual active information storage value, with a saturation point reached for certain finite k-value. Beyond this point with an increase in k there is no significant change in the finite-estimate of the information storage quantity (see Figs. 2.2 (b) and (c)).

The reservoir neuron time constants are dependent on a decay control parameter ρ_i as follows:

$$\tau_i = \kappa \left(\frac{2}{1 + \rho_i} \right)^{-m}. \tag{2.18}$$

where, $\rho_i \in \{0, 1, 2, ..., 9\}$ and κ,m are constants. Here we use $\kappa = 1$ and $m = 1.8$ such that the resultant time constants are within biologically plausible limits (Fig. 2.3)(Koch et al., 1996), (Rall, 1969).

In order to adapt the neuronal time constants τ_i the recurrent network is driven with the incoming inputs, and using epochs(ϕ time window) with finite history length $k \geq 8$, the active

---

[3] Here for mathematical convenience we represent $x_i(t)$ as $x_{i,t}$.

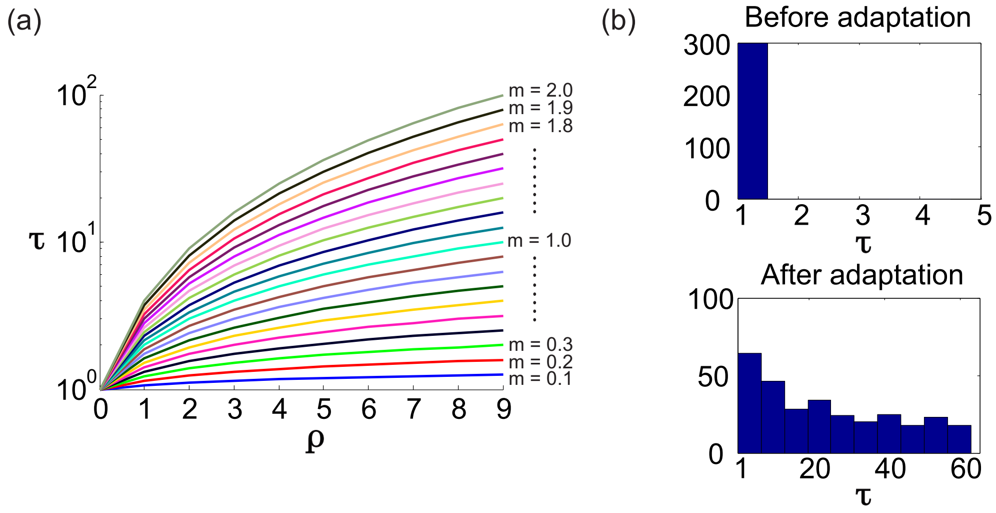


Figure 2.3: **Neuronal timeconstant adaptation** (a) plot of the range of τ values ($1-100$ ms) for a single neuron, with different parameter values of ρ and m (y-axis in log scale). (b) Example of timeconstant distribution from a reservoir of 300 neurons, (above) before adaptation (uniform distribution, $\tau = 1$) (below) after adaptation (long tailed).

information storage measure[4] at each neuron adapts the decay control parameter ρ_i as follows :

$$\rho_i = \begin{cases} \rho_i - 1 & \text{if } A_x(i,\phi) - A_x(i,\phi-1) > \epsilon \\ \rho_i + 1 & \text{if } A_x(i,\phi) - A_x(i,\phi-1) < \epsilon, \quad \text{where,} \quad \epsilon = \frac{1}{4}\log N. \end{cases} \tag{2.19}$$

After each epoch (trial), ρ_i and τ_i are adjusted and these values are used for the subsequent epoch. This procedure is typically carried out as a pre-training phase where the reservoir RNN is driven by the input signals divided into a number of training samples, over multiple trials. Once all the training samples are exhausted, the pre-training of reservoir is completed and τ_i is fixed. Given that the neuronal time constants not only act as local memory terms but as it can be observed from Eq. 2.11 it also controls the over all reservoir time-scale. Thus, our adaptation mechanism based on the change in information storage of each neuron in an input dependent manner, leads to the reservoir speeding up or slowing down its dynamics and adjusting to the timescales of the incoming input signals.

2.2.3 Homeostatic Plasticity: Information Theoretic Intrinsic Plasticity Rule

As discussed in the introductory chapter (section 1.3.2), homeostatic regulation by way of intrinsic plasticity is viewed as a mechanism for the biological neuron to autonomously modify its

---

[4] The information storage measure was implemented using modified versions of the Java based information dynamics toolkit (Lizier, 2014). The toolkit was used as a wrapper class with Matlab.

firing activity to match the input stimulus distribution (Turrigiano et al., 1994);(Desai et al., 1999). From an information theoretic perspective, Stemmler and Koch (Stemmler and Koch, 1999) demonstrated that IP can allow a neuron to exploit its complete dynamic range of firing rates while being driven by a given input, such that for Gaussian input distributions, IP could lead to an optimal exponential output distribution for maximizing information transfer (see Fig. 1.5 (c) and (d)). Furthermore, it is plausible that single neurons try to achieve this maximum information transmission while obeying constraints on its energy expenditures (Sharpee et al., 2014). Based on this idea, IP can be formalized based on the following three principles (Schrauwen et al., 2008):

- **Information maximization**: Maximum mutual information (see appendix A.1) between the input entropy of a neuron and its firing rate entropy, i.e. the output of the neuron contains as much information on the input as possible.
- **Constrained output distribution**: Neurons have a limited range of operation (firing rate range of non-linearity type) with highly sparse firing patterns as well as limits on its energy expenses.
- **Adaptation of neurons intrinsic parameters**: Biological neurons have been observed to adjust their intrinsic excitability and maintain firing rate homeostasis without the need to change individual synaptic connections (Zhang and Linden, 2003).

In (Triesch, 2007) a model of intrinsic plasticity based on changes to the neuronal non-linear activation function was introduced. A gradient rule for direct minimization of the Kullback-Leibler divergence between the neuronal current firing-rate distribution and maximum entropy (fixed mean) exponential output distribution was motivated. Subsequently in (Schrauwen et al., 2008) an IP rule for the hyperbolic tangent transfer function with a Gaussian output distribution (fixed variance maximum entropy distribution) was derived. During testing the adapted reservoir dynamics, it was observed that for temporal tasks requiring linear responses the Gaussian distribution performs well. However on non-linear tasks, the exponential distribution gave a better performance. In this thesis, with the aim to obtain sparser output codes with increased signal to noise ratio for stable temporal memory processing, we derive and implement a generic learning rule for IP using the Weibull distribution as the target output distribution, for the reservoir neurons.

The Weibull distribution is a 2-parameter continuous distribution, such that its shape and scale parameters can be adapted to account for various shapes of the neuron activation function (Eq. 2.12). The Weibull distribution has a high kurtosis number leading to sparser output codes and can generalize between a wide range of cumulative distribution functions. Unlike the previous models of fermi transfer functions (Triesch, 2007), (Steil, 2007), here we use the Weibull distribution as the target output distribution and derive a generic stochastic learning rule for tan-hyperbolic (tanh) neuronal non-linearity. This is primarily aimed at firing rate homeostasis as well as optimal information flow between the input and output of each reservoir neuron.

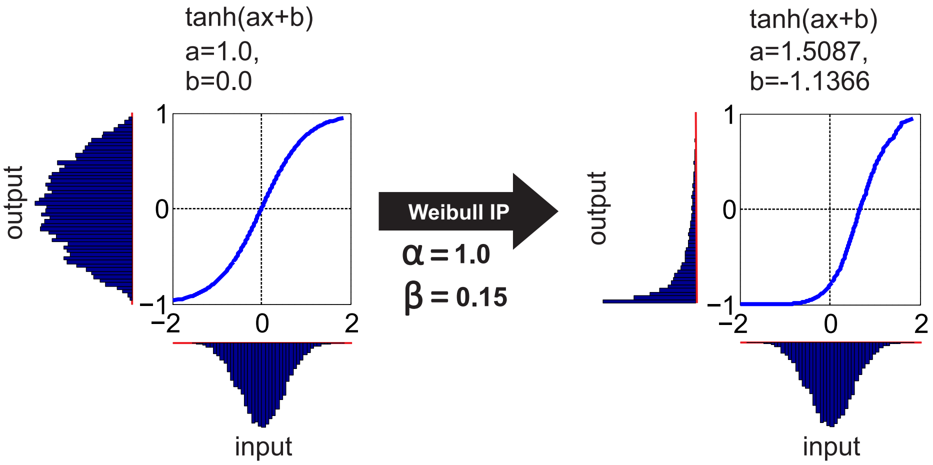


Figure 2.4: **Example of generalized Weibull intrinsic plasticity for a single reservoir neuron** (left) A hyperbolic tangent neuron firing rate function with initial shape and bias parameters, $a = 1.0$ and $b = 0.0$. Randomly selected input stimuli from a Gaussian distribution (zero mean and standard deviation 0.5) result in neuron firing rate output from a broad Gaussian distribution. (right) After intrinsic plasticity assuming an optimal Wiebull output distribution (with parameters $\alpha = 1.0$ and $\beta = 0.15$), the neuron firing rate curve shifts (learned mean value of $a = 1.5087$ and $b = -1.1366$). As a result for the same input from a Gaussian distribuition, the reservoir neuron output activity follow an maximal entropy Exponential like distribution. The Weibull distribution allows flexible adjustment of the optimal distribution shape by changing the parameters α and β accordingly.

Deriving the IP Rule for Neuron Activation Function Parameters:

The probability distribution of the two-parameter Weibull random variable r is given as follows:

$$f_{weib}(r;\beta,\alpha) = \frac{\alpha}{\beta}\left(\frac{r}{\beta}\right)^{\alpha-1} e^{-\left(\frac{r}{\beta}\right)^{\alpha}}. \tag{2.20}$$

The parameters $\alpha > 0$ and $\beta > 0$ control the shape and scale of the distribution respectively. Between $\alpha = 1$ and $\alpha = 2$, the Weibull distribution interpolates between the exponential distribution and the Rayleigh distribution. Specifically between $\alpha = 3$ and $\alpha = 5$, we obtain an almost normal distribution. Due to this generalization capability it serves best to model the actual firing rate distribution and also account for different types of neuron non-linearity. The neuron firing rate parameters a and b of Eq. 2.12 can be calculated by minimizing the Kullbeck-Leibler (K-L) divergence between the actual output distribution of the reservoir neurons activity $f_r(r)$ and the desired distribution $f_{weib}(r)$ with a fixed mean firing rate β (Fig. 2.4).

The KL-divergence between $f_r(r)$ and $f_{weib}(r)$ is given by:

$$\begin{aligned}D = D_{KL}(f_r(r), f_{weib}(r)) = &\int f_r(r) log\left(\frac{f_r(r)}{f_{weib}(r)}\right) \mathrm{d}r \\ =&\int f_r(r) log\left(\frac{f_r(r)}{\frac{\alpha}{\beta}\left(\frac{r}{\beta}\right)^{\alpha-1} e^{-\left(\frac{r}{\beta}\right)^{\alpha}}}\right) dr \\ =&\int f_r(r) log(f_r(r))\mathrm{d}r - log\left(\frac{\alpha}{\beta^{\alpha}}\right) \\ &-(\alpha-1)\int f_r(r) log(r)\mathrm{d}r + \frac{1}{\beta^{\alpha}}\int f_r(r) r^{\alpha}\mathrm{d}r \\ =&-H(r) + \frac{1}{\beta^{\alpha}}E(r^{\alpha}) - (\alpha-1)E(log(r)) - log\left(\frac{\alpha}{\beta^{\alpha}}\right).\end{aligned} \tag{2.21}$$

Here, $H(r)$ is the firing rate entropy (self-information) of a reservoir neuron.

We know,

$$H(r) = -\int f_r(r) log(f_r(r)) dr = E\Big[log\Big(\frac{\partial r}{\partial x}\Big)\Big] - E[log(f_x(x))]. \tag{2.22}$$

Using Eq. 2.22 and the relation $f_r(r) = \frac{f_x(x)}{\frac{\partial r}{\partial x}}$ (from Eq. 2.12)[5] for a single neuron with input x and output r and representing the integrals in terms of the expectation (E) quantities, the above relation can be simplified to (here C are constant terms):

$$\begin{aligned}D = -E\left[log\left(\frac{\partial r}{\partial x}\right)\right] &+ E[log(f_x(x))] \\ &+\frac{1}{\beta^{\alpha}}E(r^{\alpha}) - (\alpha-1)E(log(r)) + C.\end{aligned} \tag{2.23}$$

Recall that the tanh non-linearity can be represented in the exponential form as follows:

$$r = \mathtt{tanh}(ax+b) = \frac{e^{2(ax+b)}-1}{e^{2(ax+b)}+1} \tag{2.24}$$

Thus, differentiating this w.r.t x, a and b and representing in terms of r we get the following set of base equations:

$$\begin{aligned}\frac{\partial r}{\partial x} &= a(1-r^2), \\ \frac{\partial r}{\partial a} &= x(1-r^2), \\ \frac{\partial r}{\partial b} &= (1-r^2)\end{aligned} \tag{2.25}$$

---

[5] The activation are time dependent, however here we neglect the time variable for mathematical convenience.

Using the partial derivatives from Eq. 2.25 and differentiating D w.r.t the parameter b yields:

$$\begin{aligned}\frac{\partial D}{\partial b} &= E\left[2r + \frac{\alpha}{\beta^{\alpha}} r^{\alpha-1}(1-r^2) - (\alpha-1)r^{-1}(1-r^2)\right] \\ &= E\left[2r + r^{-1}(1-r^2)\left(\frac{\alpha}{\beta^{\alpha}} r^{\alpha} - \alpha + 1\right)\right].\end{aligned} \tag{2.26}$$

Similarly differentiating D w.r.t the parameter a results in:

$$\frac{\partial D}{\partial a} = E\left[2rx + xr^{-1}(1-r^2)(\frac{\alpha}{\beta^{\alpha}} r^{\alpha} - \alpha + 1) - \frac{1}{a}\right]. \tag{2.27}$$

From the above equations we get the following on-line learning rule with stochastic gradient descent with learning rate η

$$\Delta b = -\eta\left[2r + r^{-1}(1-r^2)\left(\frac{\alpha}{\beta^{\alpha}} r^{\alpha} - \alpha + 1\right)\right]. \tag{2.28}$$

$$\Delta a = \frac{\eta}{a} + x\Delta b \tag{2.29}$$

Note: This relationship between the neuron parameter update rules (Δa and Δb) is generic and valid irrespective of the neuron non-linearity or target probability distribution.

In general this local IP rule tries to robustly adapt the internal dynamics of the reservoir in an input driven and completely unsupervised manner. In contrast, the neural timescale adaptation rule tries to modulate the neuronal time constants, effectively matching the timescales in the incoming time varying stimuli. This is based on a quantification of the extent of influence that the past activity of a neuron has on it's activity in the immediate future. We therefore combine IP learning with the neuron timescale adaptation rule in series. The time constant adaptation is carried out after the intrinsic adaptation of the neuron non-linearity. This combination leads to a single self-adaptive framework that controls the local memory of each neuron based on the incoming input to the network, while preventing runway dynamics (homeostasis). In the next section we will present the supervised plasticity mechanism to learn the reservoir to readout and internal reservoir weights, in a task dependent manner.

2.2.4 Synaptic Plasticity: Supervised Learning and Weight Adaptation

Subsequent to the unsupervised autonomous adaptation of the reservoir neuron time constants and non-linearity parameters based on the inputs to the network, the newly learned parameters τ_i , a_i and b_i are fixed. The new network with the revised settings is now used to induce supervised synaptic plasticity (learning of connection weights) at the reservoir to readout ($\mathbf{W}^{out}$) connections and the recurrent connections ($\mathbf{W}^{rec}$) inside the reservoir.

The primary objective of weight adaptation within the framework of supervised learning is that the network learns to generate some target or desired signal $d(t)$, that may be both a function of time as well as the input to the network. The goal of supervised learning (see Tab. 1.1) is to minimize the net error function $E(T)$ between the desired signal and the actual reservoir output $z(t)$, calculated over some sufficiently long time T:

$$E(T) = \frac{1}{2}\int_0^T \mathrm{e(t)}^2 \mathrm{d}t = \frac{1}{2}\int_0^T \Big[z(t) - d(t)\Big]^2 \mathrm{d}t \tag{2.30}$$

where, e(t) is the instantaneous error signal.

Using gradient descent learning , the readout and recurrent weights can be typically calculated by minimizing this error with respect to W_i^{out} and W_{ij}^{rec}. However such an approach, based on the traditional back propagation through time (BPTT) (Rumelhart et al., 1988) learning strategy is inherently unstable and incapable of dealing with long temporal dependencies (Bengio et al., 1994) arising in large recurrent networks. Therefore, here we learn these wights using an online learning algorithm based on the recursive least squares (RLS) algorithm (Simon, 2002), which was also recently formulated as the FORCE (first-order reduced and controlled error) (Sussillo and Abbott, 2009) or FORCE-fair (Laje and Buonomano, 2013) learning setup.

Readout Weight Adaptation:

As the instantaneous error signal, $e(t) = z(t) - d(t)$, using Eq. 2.13 this can be reformulated as:

$$e(t) = \sum_j W_j^{out}(t - \Delta t) r_j(t) - d(t). \tag{2.31}$$

Using the RLS algorithm and minimize the error e^2, the readout weight (W_j^{out}) update is defined by,

$$W_i^{out}(t) = W_i^{out}(t - \Delta t) - e(t) \sum_j P_{ij}(t) r_j(t). \tag{2.32}$$

where, the error $e(t)$ is as defined in Eq. 2.31.

Here, $\mathbf{P}$ is a $N \times N$ square matrix proportional to the inverse of the correlation matrix of the reservoir neuron firing rates vector $\mathbf{r}$. $\mathbf{P}$ is initialized using the identity matrix $\mathbf{I}$ and a small constant parameter δ_c as,

$$\mathbf{P}(0) = \frac{\mathbf{I}}{\delta_c} \tag{2.33}$$

Here, $\mathbf{P}$ acts as the adaptive learning rate for Eq: 2.32, with the weight modifications automatically slowing down as $\mathbf{P}$ decreases with time. This provides inherent stability and the learning algorithm converges to a solution. **P** is updated at each time point as,

$$\mathbf{P}(t) = \mathbf{P}(t-\Delta t) - \left(\frac{\mathbf{P}(t-\Delta t)\mathbf{r}(t)\mathbf{r}^T(t)\mathbf{p}(t-\Delta t)}{1+\mathbf{r}^T(t)\mathbf{P}(t-\Delta t)\mathbf{r}(t)} \right). \tag{2.34}$$

Recurrent Weight Adaptation:

The adaptation of the recurrent weights W_{ij}^{rec} are carried out using the same supervised error signal $e(t)$ (Eq. 2.31). Based on the RLS formulation (Sussillo and Abbott, 2009), W_{ij}^{rec} is updated online as,

$$W_{ij}^{rec}(t) = W_{ij}^{rec}(t-\Delta t) - e(t) \sum_{k \in \mathbf{A}(i)} P_{jk}^{i}(t) r_k(t). \tag{2.35}$$

The notation $\mathbf{A}(i)$ represents the list of all neurons presynaptic to the neuron i. Using this notation, unlike the single inverse correlation matrix **P** of all the reservoir neurons in Fig. 2.32, $\mathbf{P}^i$ is a square matrix (one for each recurrent neuron i) with each dimension equal to the number of neurons presynaptic to i ($\mathbf{A}(i)$). This is now updated as follows:

$$P_{jk}^{i}(t) = P_{jk}^{i}(t-\Delta t) - \left(\frac{\sum_{l \in \mathbf{A}(i)} \sum_{m \in \mathbf{A}(i)} P_{jl}^{i}(t-\Delta t) r_l(t) r_m(t) P_{mk}^{i}(t-\Delta t)}{1 + \sum_{l \in \mathbf{A}(i)} \sum_{m \in \mathbf{A}(i)} r_l(t) P_{lm}^{i}(t-\Delta t) r_m(t)} \right). \tag{2.36}$$

Since the error for each of the recurrently connected neurons is the same back-propagated error from the readout neuron ($e(t)$), within this setup, we can learn the recurrent and readout weights simultaneously.

Overall Training Procedure:

The learning of the internal parameters (IP and timescale adaptation) along with the modification of synaptic weights are carried out using the following simple steps:

1. **Initialization:** The reservoir network is initialized with random parameterization. The recurrent weights $\mathbf{W}^{rec}$ are chosen randomly from a Gaussian distribution with zero mean and s.d. $g/\sqrt{p_c N}$. g is typically set between 0.9 and 1.5 and scales the synaptic strength accordingly. P_c defines the connection probability between reservoir neurons (10% to 50%). $\mathbf{W}^{in}$ and $\mathbf{W}^{fb}$ are initialized randomly from an uniform distribution over $[-1, 1]$. All neurons are initialized with $\tau = 1$, $a = 1$ and $b = 0$. The output weights $\mathbf{W}^{out}$ are either initialized to zero or chosen from a Gaussian distribution with zero mean and variance $1/N$. Depending on the complexity of the temporal information processing task the reservoir size N is selected from anywhere between 10 to 3000 neurons.

2. **Unsupervised adaptation:** The network is driven with the given time varying input signals and using a fixed number of epochs (adaptation trials) the IP and timescale adaptation procedure is carried out. After this the learned parameters τ_i, a_i and b_i are fixed for all neurons.
3. **Learning:** After this pre-training or autonomous adaptation step, learning of the synaptic connections inside the reservoir $\mathbf{W}^{rec}$ and reservoir to readout neuron $\mathbf{W}^{out}$ is carried out in a supervised manner using the target signal $d(t)$ or using a temporal difference error based on reward learning strategy (see section 5). Learning is carried out in a number of cycles, such that weights converge to a stable regime.
4. **Testing:** The learned network is now tested without any further modifications on new unseen test input signals and checked for generalization capability on the same temporal task.

2.3 Learning with Self-Adaptive Reservoir Network

In order to give an understanding of the learning and adaptation process on the ability of our network model to perform temporal information processing, we now provide an example of a relatively complex signal modeling task with inherently different timescales. Specifically we will use our SARN model to learn a two-dimensional multiple frequency sinusoidal function of the incoming inputs to the network. Typically such multiple sine problems has been very difficult to learn for static reservoir models like the echo-state network (Jaeger, 2001a) or their spiking neuron counterparts, liquid state machines (Maass et al., 2002). Therefore, we take this example in order to display the learning behavior of our network and compare it to the performance of an optimized static reservoir network (without any internal adaptation) (Jaeger et al., 2007). Further elaborate results of temporal processing and memory guided behaviors will be provided in chapter 3.

In this setup, the network was stimulated by a two dimensional input time series $u_1(t)$ and $u_2(t)$ drawn from an uniform distribution in the closed interval $[-1, 1]$. The goal of the modeling task was to learn the following function of the inputs:

$$f(u) = sin(\omega\pi(u_1^2 + u_2^2)) \tag{2.37}$$

where, the parameter ω controls the frequency of the mapping. This was steadily increased in value such that, $\omega \in \{1, 2, 3, ..., 10\}$, and the network needs to learn all the 10 frequencies at the same time.

The network was randomly initialized with default parameters as explained in the section 2.2.4. It consisted of $N = 100$ neurons scaled with a factor $g = 1.1$, along with two input neurons and ten output neurons (one for each frequency ω of sinusoidal signal). In order to carry out IP adaptation the Weibull distribution was initialized with parameters $\alpha = 1.0$ and $\beta = 0.3$.

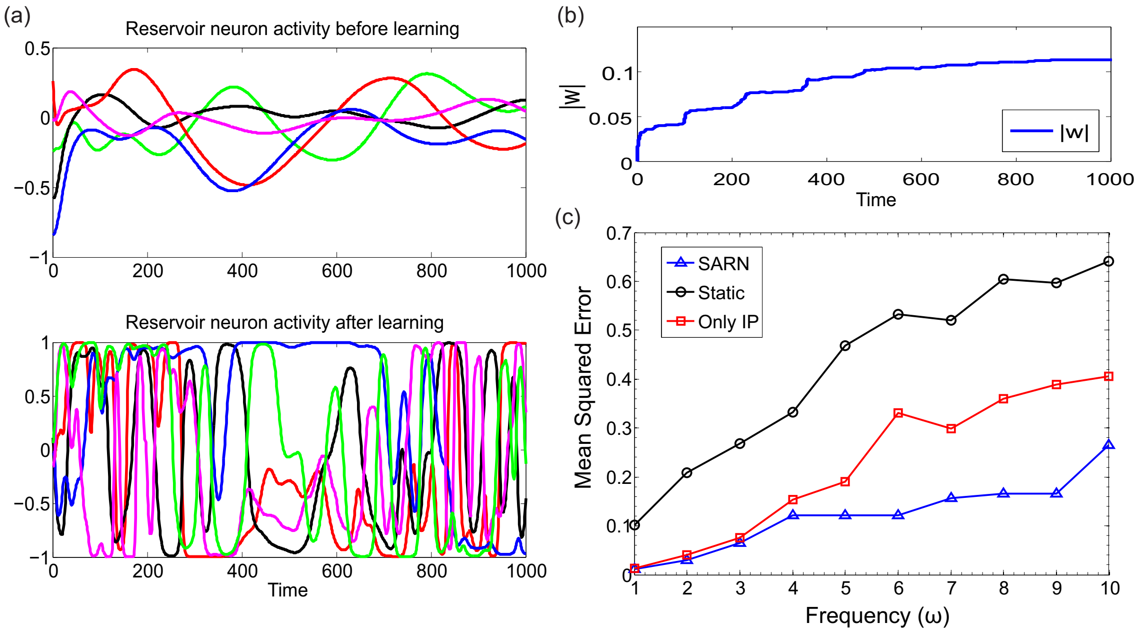


Figure 2.5: (a) Activities of five randomly selected reservoir neurons are shown. Above - reservoir activity resembles a single slowly varying sine function. Below - after supervised learning, reservoir activities display diverse signals with clearly two different embedded time scales of sine function(b) Change in length of the readout weight vector $|\mathbf{W}^{out}|$ during training (c) Plot of the mean squared error (mse) of the network output after training, with respect to changing ω values. The blue curve shows the performance of the current SARN model with both IP and timescale adaptation; red curve shows SARN with only IP optimization and black curve shows performance of the static un-adapted reservoir network.

During the pre-training phase, using 300 epochs, IP and timescale adaptations were carried out. After this the learned neuronal time constants and neuron non-linearity parameters were kept fixed and synaptic modifications as per Eqs. 2.32 and 2.35 were carried out. As observed in Fig. 2.5 (a) above , before learning the reservoir network showed regular single sinusoidal activity. However after the network had been trained (Fig. 2.5 (b) below) and adapted for the task, the reservoir neurons now encoded two distinct timescales in their activity. A slowly varying intermediate transient with fast changing signals at the two extremes is clearly observed in the activity of the selected reservoir neurons. As a result the SARN model was able to almost perfectly learn the desired output sine signals with ten different frequency components (Fig. 2.6 (a)).

In order to measure the stability of the learned behavior we calculate the length of the readout weight vector $|\mathbf{W}^{out}|$ and plot the change in its value with time, in Fig. 2.5 (b). Large values of $|\mathbf{W}^{out}|$ typically indicate that the solution found by a learning process involves cancellations between large positive and negative contributions that tend to be unstable and sensitive to noise (Sussillo and Abbott, 2009).

As observed, in this case, the overall magnitude of $\mathbf{W}^{out}$ remains relatively small. Furthermore, during the training, although there is an initial increase in $|\mathbf{W}^{out}|$. After some time period, it reaches a stable region (plateau) indicative of the learning completion. The performance of SARN model was measuring by calculating the mean squared error between the learned outputs ($z_i(t)$) and the actual desired output function (Eq. 2.37) for each value of ω.

Furthermore we tested the performance of a static reservoir model and a reservoir with only IP for the same task. As observed in Fig. 2.5 (c) with an increase in the frequency the performance drops significantly. However the SARN model clearly outperforms the static reservoir, where a rapid deterioration of performance is observed for $\omega \geq 2$. Intrinsic plasticity can be seen to help in the function approximation process however, only IP adaptation still leads to large errors for higher frequency components; on the contrary with both IP and timescale adaptation the error in the output is considerably reduced with negligible change in the MSE for $4 \leq \omega \leq 9$. This can be attributed to the slow and fast dynamics needed to approximate the output patterns, which is achieved by the combination of different neuronal time constants learned in the timescale adaptation process. Figure 2.6 (b) and (c) shows the learned output (for two frequency components) as compared to the desired signal. SARN can be seen to reproduce the output near perfectly as compared to the irregular output from the static network. Thus using this experiment, we demonstrate how the SARN model can be used to learn a basic temporal processing task as well as the performance benefit obtained by the homeostatic IP and local neuron memory (time constant) adaptations.

2.4 Summary

In this chapter, based on the framework of input-driven RNN, we presented the detailed description of the self-adaptive reservoir network (SARN). Using an information theoretic approach we introduced novel learning rules for autonomous adaptation of reservoir neuron membrane time

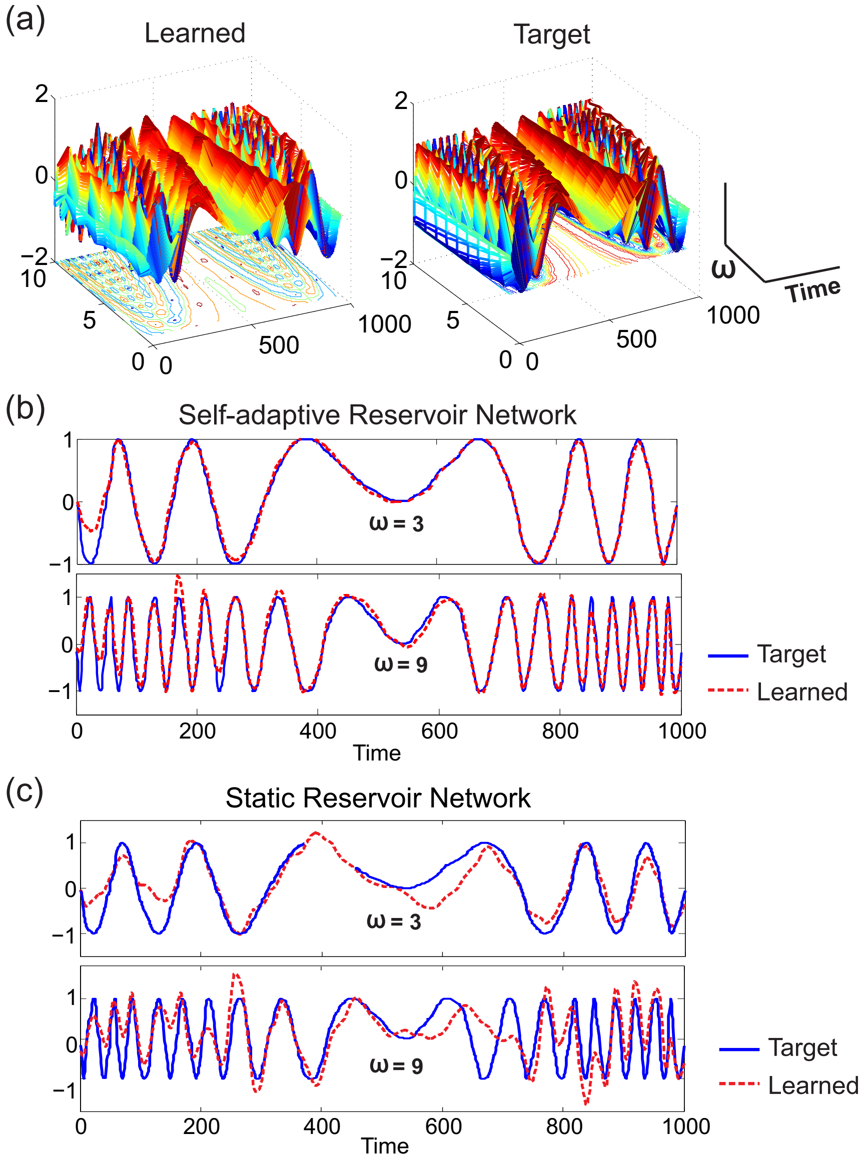


Figure 2.6: **Multiple frequency sine modeling task with SARN** (a) Comparison of learned output from SARN as compared to the actual desired response. Output from all ten readout neurons are plotted as a function of time and ω (ω encodes the frequency component of Eq. 2.37) (b) Response of SARN outputs for $\omega = 3$ and $\omega = 9$. The reconstruction performance visibly drops with the increase in frequency, however SARN is still able to model the correct response with considerable accuracy as observed from the overlap of learned and target signals. (c) Response of the static reservoir output neurons for the same frequency components. As observed, the reconstruction accuracy is very poor with the output activity failing to learn both the slow and fast components of the signal, even for small frequencies ($\omega = 3$).

constants, along with an online stochastic intrinsic plasticity (IP) rule to adapt neuron activation function parameters. The ability of RNN to perform brain like temporal information processing, is contingent on the robust multiplexing of these time varying stimuli. Here we consider that temporal multiplexing and adaptation to varying timescales of inputs can arise through biologically plausible mechanisms of IP and local neuron memory modifications. We introduced a novel input-driven active information storage rule that can be used to modify the decay rates of individual reservoir neurons. This allows the neurons to locally quantify the information dependence between their past and future states, given the context of the inputs currently driving them. Since membrane time constant or decay can be viewed as local memory terms, this enable the reservoir neurons to adjust the memory of their dynamics or speed, according to the timescales in the incoming input. In addition, we derive a generic IP rule based on Weibull probability distribution that ensures maximum flow of information between the input and output of each reservoir neurons and maintain homeostasis in the network. Finally we describe a supervised synaptic plasticity rule that can be used to learn the strength of synaptic connections both, inside the reservoir as well as from the reservoir-to-readout neurons in an online and stable manner. This learning mechanism and the increased performance of our SARN model in comparison to optimized but static reservoir networks is clearly demonstrated using the example of a multiple frequency two dimensional sine function generation task. In the next chapter we will present detailed experimental results of using SARN within a closed loop paradigm, by evaluating it on various complex temporal processing tasks, from synthetic time series data to generating delay and sequence memory guided behaviors in artificial agents.

CHAPTER 3

Robust Temporal Information Processing with Self-adaptive Reservoirs (Experiments and Results)

> ”Time present and time past are both perhaps present in time future and time future contained in time past.”
>
> —*T.S. Eliot*

In the previous chapter we provided the necessary theoretical background and methodical details of the self-adaptive reservoir network (SARN). Furthermore, it was also clearly demonstrated, using a preliminary multiple timescale sinusoidal modeling task, that SARN, based on the intrinsic plasticity, reservoir neuron timescale adaptation and supervised synaptic plasticity, significantly outperforms static, non-adaptive networks. In this chapter, we further elaborate on this result, using experimental setups that reflect temporal information processing and memory guided behaviors of different degrees of complexity, in the timescale of few milliseconds to minutes. We start with a number of standard benchmark, synthetic time series processing tasks and evaluate the performance of SARN in comparison to its static reservoir counterparts. We also demonstrate that, unlike current recurrent neural network models, SARN is able to encode both stable and chaotic attractors in its dynamics and make robust predictions based on it, in an online manner. Using Lyapunov stability analysis, we show that the plastic and adaptive mechanisms in SARN lead to a more near critical network, as compared to previous RNN models that are either highly chaotic or sub-critical, for the same network scaling parameter settings. This is followed by delay temporal memory and timing based based experiments on an artificial walking robot using SARN in a closed loop approach. Finally we demonstrate the use of SARN for a complex motor processing task like handwriting generation (under perturbations for a multiple degree of freedom robotic arm), and compare its performance to two of the most recent state of the art static chaotic RNN (Sussillo and Abbott, 2009) and plastic (innate learning) RNN (Laje and Buonomano, 2013).

3.1 Synthetic Time Series Processing

3.1.1 Benchmarking on Standard Datasets

The performance of our self-adaptive reservoir network in processing of complex time-varying information, is evaluated using three standard benchmark time series data (Schrauwen et al., 2008), (Jaeger, 2001a), (Jaeger, 2001b), (Jaeger and Haas, 2004), (Steil, 2007), (Rodan and Tino, 2011), covering a wide spectrum of temporal structure (multiplexed timescales), non-linearity and memory. In all the cases, we compared the performance of SARN with that of an optimized version a static reservoir network (i.e. without any internal unsupervised adaptation). In the following sections we will now describe the experimental setup of the reservoir, followed by a brief description of each task.

Experimental Setup

In all experiments in this section, the internal reservoir network weights W^{rec} were initially drawn randomly from a normal distribution with zero mean and standard deviation $(g^2/\sqrt{p_c N})$. The network size N was either fixed at 300 neurons or varied between $100 - 400$ neurons, initialized with a connectivity of 20% i.e. $p_c = 0.2$. The reservoir network was scaled using $g = 1.2$. The input $\mathbf{W}^{in}$ and feedback weights $\mathbf{W}^{fb}$ were drawn randomly from a uniform distribution $[-0.5, 0.5]$. The reservoir neuron firing rate parameters were initialized with $a = 1$ and $b = 0$. The learning rate of the stochastic gradient descent algorithm was fixed at $\eta = 0.0001$. Neuronal time constants were initialized to $\tau = 1ms$. Intrinsic plasticity and reservoir neuron timescale adaptations were carried out in 100 epochs using overlapping time windows of 1000 time steps. Subsequently the non-linearity parameters and time constants were fixed for the supervised learning of reservoir recurrent and reservoir-to-readout weights.

Dynamics system modeling with 15th order NARMA

The dynamics of the n^{th} order non-linear auto-regressive moving average (NARMA) is given by:

$$d(t+1) = 0.2d(t) + 0.004d(t)\sum_{i=0}^{n-1} d(t-i) + 1.5u(t-(n-1))u(t) + 0.001 \tag{3.1}$$

Here $n = 15$ for the 15th order modeling scenario and $d(t)$ is the output of the system at time 't'. $u(t)$ acts as the input to the system at time 't', and is uniformly drawn from the interval [0,0.5]. The task is to output $d(t)$ based on $u(t)$. In general this task is quite complex considering that the current system output depends on both the current time step input as well as its own previous $n-1$ time steps history. Consequently, we use feedback connections ($\mathbf{W}^{fb}$) from the output neurons to the internal neurons with the reservoir network dynamics evolving according to Eq. 2.14. Due to this inherent dependence on own previous history this task requires extended temporal memory with the complexity increasing with higher orders of the system. The training, and testing were carried out using 1000 and 3000 time steps respectively. The network setup consisted of a single input neuron, feeding the input $u(t)$ to the reservoir network and just one output neuron trying to model the desired signal $d(t)$.

Sante Fe laser data prediction

The Santa Fe laser data (Jaeger and Haas, 2004) is a cross-cut through periodic to chaotic intensity pulsations of a Far-Infrared-Laser in a chaotic state. The chaotic pulsations more or less follow the theoretical Lorenz model of a two level system (Huebner et al., 1989). The main task is to predict the next laser activation $d(t+1)$, given the values up to time t (a small fragment of the actual data is shown in Fig. 3.1). Due to the intermixing of periodic and chaotic fluctuations, this data inherently contains multiple timescales making the prediction task quite complex. The original dataset contained 10000 data points. Here the first 6000 were used for training and then the learned network was tested with the remaining 4000 time steps of data.

Delayed n-bit parity task

The delayed n-bit parity task functions over input sequences t time steps long, and determines for n bits, if $\tau_d + n \rightarrow \tau_d$ time steps in the past are active. Here τ_d represents the delay period. The input consists of a temporal signal $u(t)$ drawn uniformly from the interval [-0.5,0.5]. Using $n = 3$ bits, the desired output signal is calculated as the PARITY function:

$$d(t) = u(t-\tau_d) \oplus u(t-\tau_d-1) \oplus u(t-\tau_d-2) \tag{3.2}$$

with increasing values of time delay (τ_d), such that $0 \leq \tau_d \leq 400$. Here, $\oplus$ is the logical XOR operation.

Since the parity function (XOR) is not linearly separable, this task is quite complex and requires both the computational ability to perform a parity check, as well as the ability to recall long spans of the input signal (fading memory). The network setup consisted of a single input neuron, the internal reservoir network with 400 neurons and 400 readout neurons (each readout neuron represent the 3-bit parity for each value of $\tau_d \in [0, 400]$).

Here, we evaluated the delayed short-term memory capacity (MC_{τ_d}) of the network as the amount of variance of the delayed input signal that is recoverable from the optimally trained readout neurons for different time delays (τ_d). This measure was first introduced by (Jaeger, 2001a), and has been subsequently adopted as a standard measure of memory capacity for

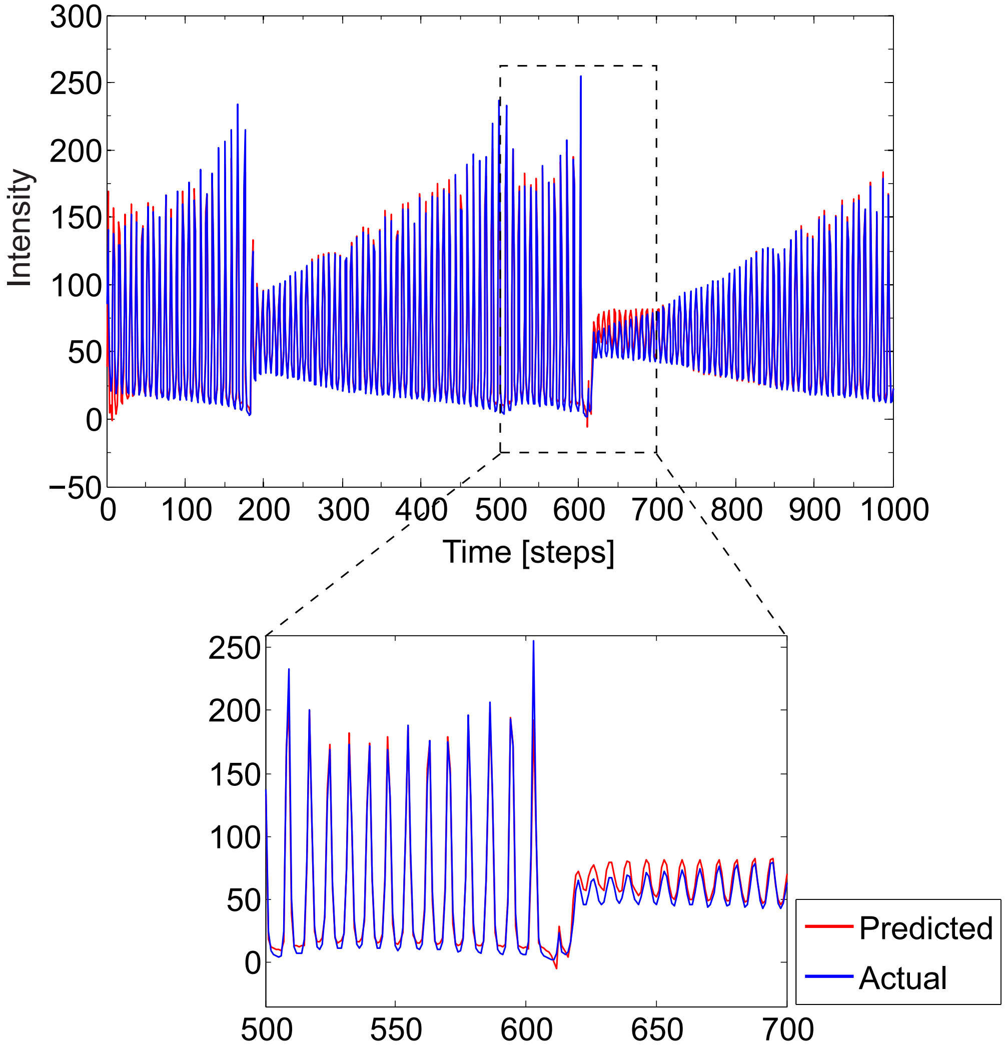


Figure 3.1: **Santa Fe laser data prediction of periodic and chaotic fluctuations** A fragment of the actual laser data (blue) along with the SARN predicted output(red), showing the multiple timescales present in the data. The inset shows a zoomed in view of a transition zone (between 500 to 700 time steps) from periodic oscillation to chaotic fluctuation. There is evidently a drop in the prediction accuracy at the transition point, however with a fast recovery to the original signal. Similar trends can be observed throughout the entire data.

reservoir networks (Lukoševičius and Jaeger, 2009), (Ganguli et al., 2008). For a given input signal delayed by τ_d time steps, the delayed memory capacity is given by:

$$MC_{\tau_d} = \frac{cov^2(z(t-\tau_d), d(t))}{var(z(t))var(d(t))} \tag{3.3}$$

where cov and var denote co-variance and variance operations, and $z(t)$ and $d(t)$ represent the reservoir actual output signal and desired signal, respectively.

The total amount of memory present in the reservoir network can be quantified by summing over all the time delays:

$$MC = \sum_{\tau_d=0}^{400} MC_{\tau_d} \tag{3.4}$$

The performance of both SARN and static reservoir networks in modeling or predicting the desired signal $d(t)$ in the cases of NARMA-15 and laser data, were evaluated using a normalized mean squared error (nmse) between the desired signal $d(t)$ and the actual network output $z(t)$, i.e. nmse $= \left(\frac{\left\langle (d(t)-z(t))^2 \right\rangle_t}{\left\langle (d(t)-\left\langle z(t) \right\rangle)^2 \right\rangle_t} \right)$.

Results

In Fig. 3.1, we plot a fragment of the Santa Fe laser data and the predicted output of the SARN network. Visual inspection depicts that SARN was able to predict the chaotic fluctuations of the data with significant accuracy. The abrupt changes in timescale of the data is apparent in the Fig. 3.1 inset (between 500 to 700 time steps). Although there is evidently a relatively large error at the transition point, the learned signal quickly settles down on the correct trajectory and predicts the remaining data points with near perfect accuracy. In order to further quantify the prediction performance of SARN, we tested the same task with increasing network size from $N = 100$ to $N = 300$ neurons and compared the performance with a static reservoir network. All the parameters of the compared static reservoirs were set to their critical values (through empirical parameter scanning), such that they operated at their optimal level of performance. As observed in Fig. 3.3 (a), the SARN network leads to much smaller values of prediction error (nmse) and an expected increase in performance with increasing network size. However, unlike the static reservoir network, due to the internal adaptations in SARN there is no significant change in the prediction error for $N > 200$ neurons, thus leading to a more stable and robust performance. Given that the laser data contains multiple periodic and chaotic fluctuations overlapped in a single time-series , it inherently has many timescales or frequency components (see Fig. 3.1 fast and slow changing regions). As such, we hypothesize that the robust performance of SARN as compared to the static network, in this case primarily arises due to the reservoir neuron timescale (decay rate or timeconstant τ) adaptation mechanism. In order to investigate this

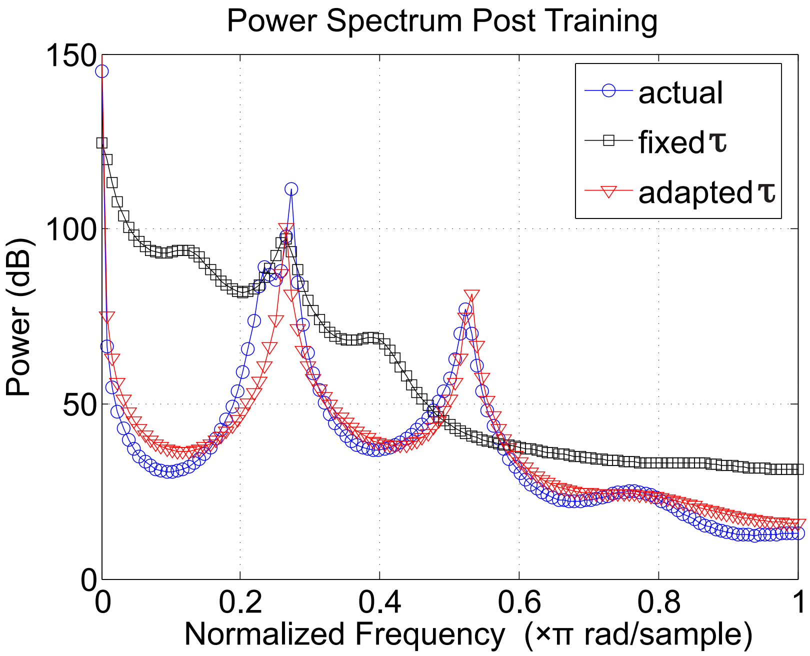


Figure 3.2: **Power spectral density estimate of laser data after training.** SARN wihtout reservoir neuron timescale adaptation (variable decay rate τ) fails to match the intensity-frequency relationship of the actual laser data. Due to the periodic to chaotic fluctuations, certain low frequency components can be seen to have much higher power as compared to others. SARN with adapted τ captures this relationship significantly well, leading to the low prediction error and robust performance.

further, we carried out the same task with SARN keeping all parameters the same as above, however now with a fixed τ for each reservoir neuron (essentially timescale adaptation was switched off, however IP functions normally). To compare the performance of SARN with fixed τ and SARN with adapted τ, we calculated the power spectrum density (PSD) of the predicted outputs of each network and compared it to the PSD of the actual laser data. As observed in the Fig. 3.2, as expected there exists multiple frequency components in the original signal, with significant variations in the loudness (power). Clearly, PSD of output signal from SARN with timescale adaptation matches that of the original signal near perfectly. However, for SARN without timescale adaptation, the PSD of the predicted output is much louder on average for lower frequencies, with a steady drop in power as the frequency increases. As such without the adaptive τ, the reservoir output significantly failed to learn the different fast and slow temporal (normalized frequency) aspects of the Laser data.

Similar trend as the performance with the laser data, is also visible in case of the NARMA-15 task (Fig. 3.3 (b)). Once again the plastic and adaptive changes in SARN lead to a much higher performance (low nmse values) while comparing same size static networks. Interestingly, although SARN outperforms static reservoirs here, the change in performance is not as pronounced as the laser data case, with considerable difference only in larger network sizes. This

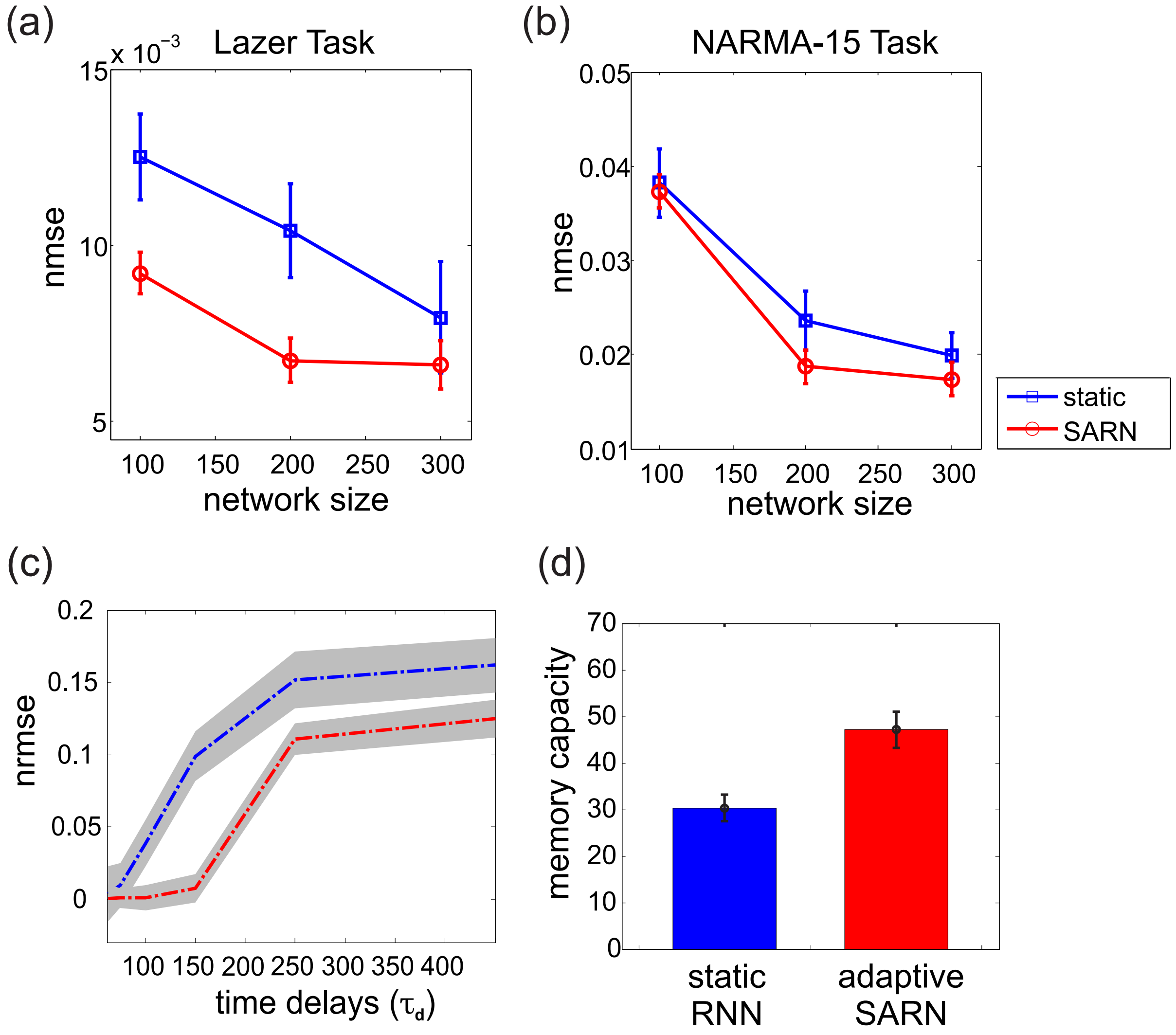


Figure 3.3: **Comparision of learning performance and memory capacity of SARN and a static RNN (reservoir) on the three benchmark tasks** (a) Performance (mean nmse values) of SARN as compared to a static reservoir network for the Santa Fe lazer prediction task plotted as a function of the reservoir network size. Improvement in the nmse values can be observed using SARN with increasing network size. (b) Mean nmse values plotted against the network size for SARN and static reservoir for the NARMA-15 task. In both (a) and (b) error bars indicate standard deviation accross 10 different trials. (c) Plot of normalized root mean squared error (nrmse)on the delayed 3-bit parity task for increasing delay (τ_d) values, comparing a 400 neuron SARN (in red) with a same size static network (in blue). SARN retains a longer memory, robust upto long delay spans as indicated by the lower nmse values. Grey shaded regions indicate the standard deviation of error values accross 10 trials. (d) Comparrision of total memory capacity as calculated by equation 3.4 across all delays. SARN acheives a high MC of 47.173 ± 2.831, while the MC in static reservoir was considerably low at 30.362 ± 2.793, for the same size network. Previous methods could acheive such long memory spans only in case of specifically designed network topologies (Boedecker et al., 2009).

can be attributed to the timescale adaptation of single reservoir neurons, that enable SARN to robustly encode the multiple timescales in laser data. In case of NARMA, the task mainly required a high degree of non-linear computation capability, which can be achieved by the combination of IP and a large network size allowing diverse reservoir signals to be present.

The delayed 3-bit parity task requires an inherently long fading memory of the incoming random input signals in order to compute delayed versions of it. As such, we used this task in order to compare the performance of our adaptive reservoir network with static reservoirs whose parameters were optimized offline in a task specific manner. As observed from the normalized root mean squared error (nrmse) curves (Fig. 3.3 (c)) for the parity values calculated for each time delay (τ_d), the SARN significantly outperforms static networks, specially for long delay times. Furthermore, due to the random initialization of the static network, it tends to have a much higher standard deviation of error (grey areas) and consequently was less robust across all trials. We further quantified the total memory capacity of each network across all time delays (Eq. 3.4) from 10 different trials. The SARN achieved a particularly high mean memory capacity of 47.173 with standard deviation 3.831, while the static reservoir network had a mean capacity of 30.362 with standard deviation 2.793. As such with adaptation, for the same network size, an increase by $\approx 22\%$ in the net short-term memory capacity of the network was observed. Previously, non-normal networks (e.g. a simple delay line network) have been shown to theoretically allow extensive memory (Ganguli et al., 2008) which is arguably not possible for arbitrary recurrent networks. However our self-adaptive reservoir network shows considerable increase in the memory capacity (with a fixed network size of 400 neurons), which was previously shown to improve only in case of specifically designed network topology (permutation matrices as internal network weight configurations) (Boedecker et al., 2009). Overall these results clearly indicate the increased performance of SARN for temporal information processing for both non-linear computation power as well as dealing with relatively long time spans of input history.

3.1.2 Multiple Attractor Pattern Recognition

In the previous subsection, based on standard benchmark tests, we clearly demonstrated that the plastic mechanisms in SARN lead to a considerable increase in performance as compared to static reservoir networks. These time series processing tasks reflect in general, various degree of complexity in terms of non-linear computation, multiple timescale adaptation and temporal memory, needed for brain like temporal information processing. Here we will now investigate further, the effect of the plastic adaptation mechanisms introduced in this thesis on the ability of the reservoir network to transiently hold both stable and fragile time-varying patterns, and be able to selectively recall or recognize them in an input driven manner.

In order to generate these patterns we make use of the well known Mackey-Glass non-linear time delay differential equation, which can have complex dynamics including stable periodic to chaotic attractors (Mackey et al., 1977). Unlike low dimensional dynamical systems, such as the Lorenz equation (Lorenz, 1963) and the Rössler equation (Rossler, 1979), the Mackey-Glass equations are infinite dimensional systems, wherein changes in its parameters lead to bifurcations in its dynamics. These have been related to the complex dynamics observed in physiological processes in biological systems (Glass and Mackey, 1988) and as such forms an ideal setting for generating multiple attractor patterns, to test the performance of SARN to robustly encode such complex and temporally intricate dynamics.

The general form of the Mackey-Glass time delay equation is as follows:

$$\dot{d}(t) = \beta_m d(t) + \frac{\alpha_m d(t-\tau_m)}{1 + d(t-\tau_m)} \tag{3.5}$$

where, $\beta_m = -0.1$ and $\alpha_m = 0.2$. Here the parameter τ_m defines the amount of time of delay in the system and for $\tau_m > 16.8$ it displays high dimensional chaotic attractors.

Here, using $\tau_m = 5$ and $\tau_m = 9$ we generated two stable periodic time series and, using $\tau_m = 17$ and $\tau_m = 28$ we generated two mildly chaotic and highly chaotic time series data, respectively. These were then loaded into the reservoir network (both SARN and a static version of the network as before) as four input patterns. In addition, four different context signals were provided as additional inputs in a 1-of-4 encoding (given as brief 100 ms pulse input to the network), such that, only one of the context signals were active at a time. The task was designed such that, once the reservoir is loaded with both the stable and fragile (chaotic) patterns, depending on which of the context signal is active, it needed to learn to generate the respective time series pattern accurately for a certain period of time (i.e. learn the respective stable or chaotic attractor). As pointed out recently in Jaeger (2014), as well as from previous attempts to model chaotic time series data (Jaeger and Haas, 2004), it is known to be non-trivial in the first place to train an RNN to stably generate any one of these patterns. However, here we loaded both stable and unstable attractor patterns into the same reservoir, and learn to generate all in a context dependent manner. As such, in order to learn this task , the network needs to be able to encode both stable and chaotic attractors in its internal dynamics.

Here we used a network of size $N = 1000$ neurons with eight inputs (four time-varying patterns and four context signals) and two readout neurons representing the generated pattern ($z(t)$) along with its time delayed version $z(t - \tau_m)$. The network was initialized using the same parameters as introduced in the experimental setup in the previous subsection, however, here we used an initial network scaling factor of $g = 1.5$, such that the network activity showed spontaneous chaotic dynamics (Sompolinsky et al., 1988). Pre-training of the SARN network was carried out using 50 epochs of four Mackey-Glass time varying patterns with the different τ_m values. After this, the network non-linearity and time constant parameters were fixed and plastic changes of the internal recurrent connections and reservoir-to-readout connections were carried out. In the static reservoir case, no pre-training took place, and the randomly generated network was directly trained using supervised learning. The original signal $d(t)$ and its delayed version $d(t - \tau_m)$ was used as the training signal during the learning process in all cases.

In Fig. 3.4 we plot the four different delay embedded versions of the Mackey-Glass patterns and the outcome of training with the SARN and static reservoir network. As observed in Fig. 3.4 (b), the SARN network robustly learns to generate both, stable and chaotic attractors, visibly similar to the original pattern as generated from Eq. 3.5, Fig. 3.4 (a). Depending on the current

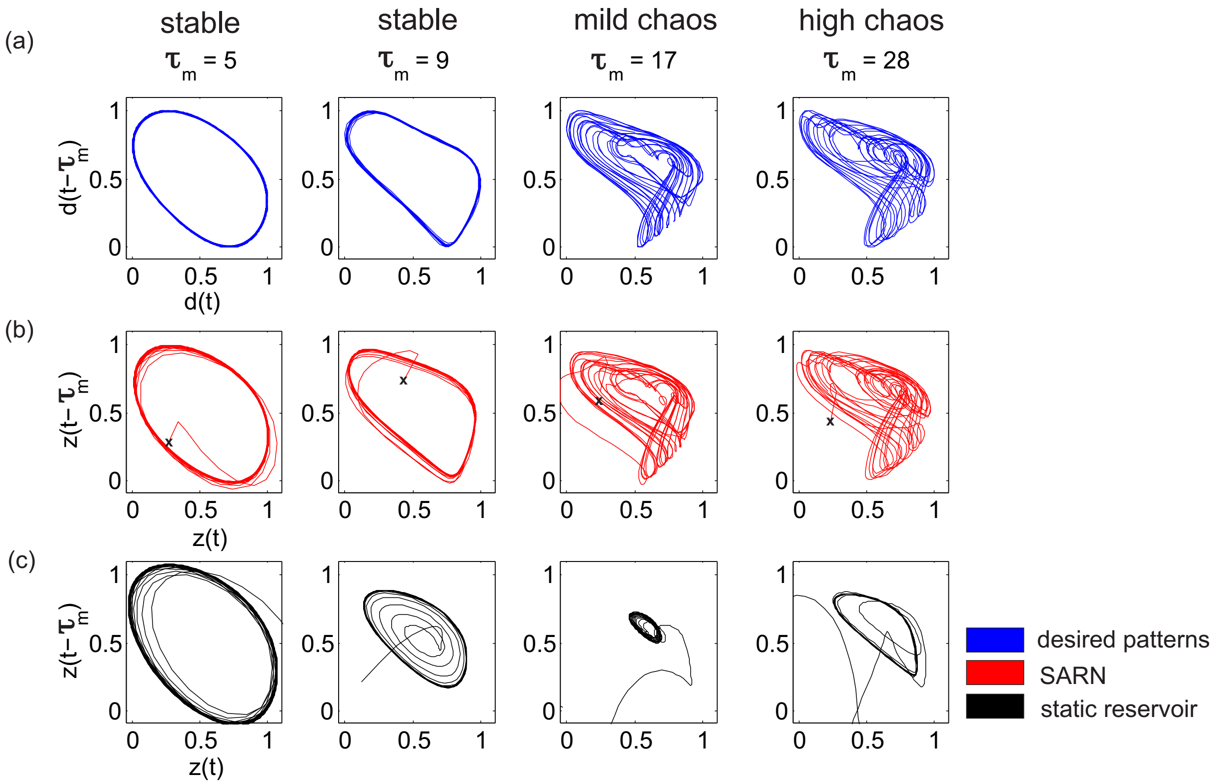


Figure 3.4: **Time delay embedded plots of the different stable and chaotic Mackey-Glass attractors learned by the self-adaptive reservoir network as compared to a static reservoir** (a) Original stable periodic attractors ($\tau_m = 5, 9$) and high dimensional chaotic attractors ($\tau_m = 17, 28$) generated by Eq. 3.5. (b) Attractors learned by the SARN network. Here 'x' markes the starting point of the learned trajectory. Depending on the contextual input, any one of these output are active at a time. Visual inspection shows that the learned attractor pattern is satisfactorily close to the original pattern above. (c) Learned attactor patterns for the static resservoir. The stable periodic attractors resemble the original patterns to some degree, however the network seems to get stuck in a limit cycle for high τ_m values, and is unable to learn the chaotic attractors.

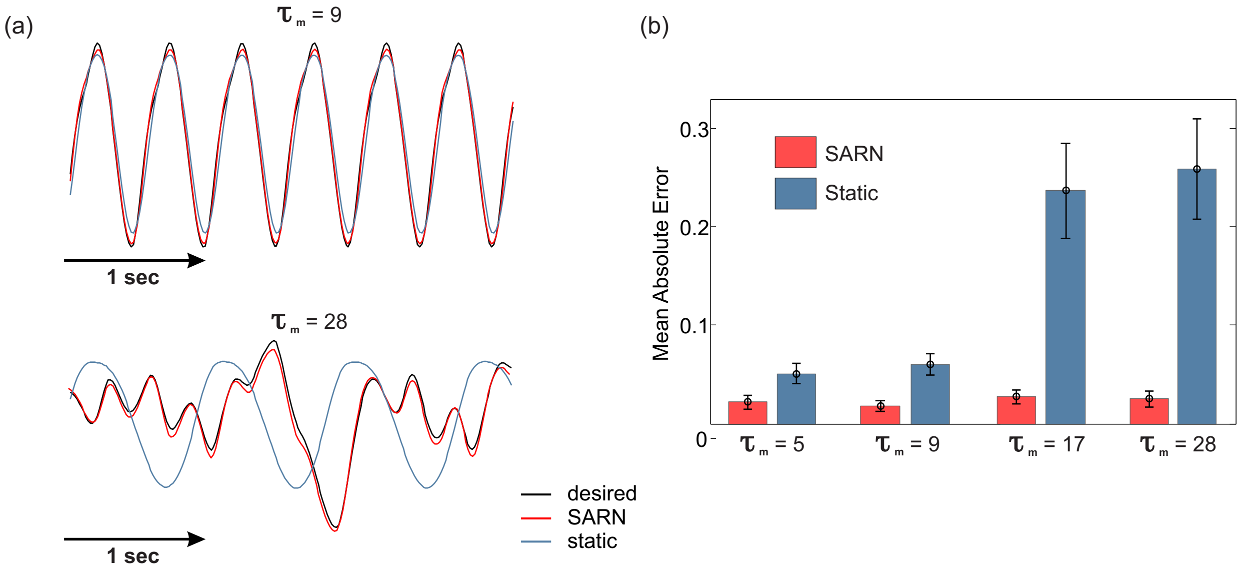


Figure 3.5: **Performance comparrison between SARN and static reserovir for the multiple Mackey-Glass attractor generation task** (a) (above) section of the learned output tragectories for SARN and static reservoir compared to the actual signal for stable attractor case $\tau_m = 9$. (below) chaotic attractor case $\tau_m = 28$. (b) Mean absolute error between the generated patterns and the desired pattern for all four cases comparing SARN and static networks. Bars indicate mean values accross 10 different trials and error bars indicate standard deviation with 95% significance level.

context input the network generates one of these patterns as output $z(t)$ and $z(t-\tau_m)$, starting from the location 'x' in the phase space. Given that the network consists of 1000 neurons, the reservoir network dynamics are embedded in a 1000-dimensional state space. Depending on the context input, the network dynamics follows a particular trajectory along this 1000-dimensional space leading to the corresponding output trajectory (see subsection. 3.1.3 for input specific Lyapunov exponent analysis). However, in comparison, the static reservoir is unable to generate all the four patterns. Visual inspection of Fig. 3.4 (c), shows that the static network learns the stable attractors to some degree of accuracy, however performs poorly in generating both the chaotic attractors. Furthermore, from the observed pattern of the learned chaotic trajectory, the network dynamics seems to be stuck in a limit cycle of the stable periodic attractor and the context inputs are unable to push the dynamics towards the chaotic domain, and the outputs continue to generate a periodic pattern. This is further illustrated in the time-series segment shown in Fig. 3.5 (a), demonstrating the learned outputs for time delays $\tau_m = 9$ (stable) and $\tau_m = 28$ (chaotic).

In order to further evaluate the performance of both the reservoir networks, we carried out ten different trials with random weight initializations for both SARN and the static network, and recorded the mean absolute error (MAE) between the reservoir output $z(t)$ (Eq. 2.13) and the desired Mackey-Glass output $(d(t))$ for the specific time delay (τ_m). As observed in Fig. 3.5 (b) SARN outperformed the static reservoir in all the four patterns. In case of the stable attractors for $\tau_m = 5$, SARN recorded a considerably low MAE of 0.0220 ± 0.0072 and for $\tau_m = 9$ an MAE of 0.0175 ± 0.0054 was observed. Here however, the static reservoir network performance was also relatively good for these stable patterns with an MAE of 0.0507 ± 0.0101 and 0.0601 ± 0.0112 for the first and second delay times, respectively. However, there was a significant difference in error for the chaotic patterns, with MAE of 0.0274 ± 0.0069 for SARN and an MAE of 0.2370 ± 0.0483 for the static reservoir ($\tau_m = 17$). The difference in error was even larger for the highly chaotic pattern ($\tau_m = 28$), with an MAE of 0.02550 ± 0.0080 for SARN and an MAE of 0.2589 ± 0.05108 for the static network (showing a performance drop of $\approx 82\%$). Thus although the static networks were able to learn the periodic patterns well, they failed to learn both the mildly and highly chaotic patterns. Clearly plasticity and adaptation mechanisms (combination of IP and neuron timescale adaptation) in SARN, allow the existence of both stable and very fragile attractors inside the reservoir dynamics such that the respective output can be generated in a robust manner. The static network dynamics on the other hand seem to get entrained to a stable domain resulting in periodic output patterns, even when the context input signals were changed (note that both the networks showed chaotic internal activity in the absence of inputs owing to the recurrent weights scaling with $g = 1.5$).

3.1.3 Near Critical Dynamics: Largest Lyapunov Exponent Analysis

In order to formally characterize the dynamics of the networks and also check the influence of the context input signals on the underlying dynamics, we estimated the largest Lyapunov exponent (LLE - λ) of the network trajectories before and after training for both the reservoirs. LLE provides a measure of the rate of separation of two nearby points in the network state space and provides a standard approach for determining if a dynamical system is chaotic (Laje and Buonomano, 2013). Although traditionally LLE analysis is designed for the characterization of autonomous dynamical systems (ergodic), previous work (Jaeger and Haas, 2004), (Rodan and Tino, 2011), (Laje and Buonomano, 2013) has shown that it can be extended to input-driven dynamical systems (recurrent neural networks of the reservoir type). We estimated the local LLE λ (finite time estimation, see appendix A.2) using a procedure similar to the estimation of local divergence rates of nearby trajectories from finite time series data as demonstrated in (Kantz, 1994), (Sprott and Sprott, 2003) and extended to RNNs in (Jaeger and Haas, 2004). If the estimated value of λ is positive and greatly bigger than zero, it means that the perturbations in the network are amplified (locally diverging trajectories) causing the network to be in a supercritical or chaotic dynamical regime. If it is negative, then perturbations are attenuated (locally contracting trajectories) causing network to be subcritical or highly stable dynamical regime. However for a λ value equal or very close to zero the network exhibits critical dynamics (marginal stability) and is said to be on the so called ”edge of chaos” (Legenstein and Maass, 2007b).

From the ten trials for the learning of the Mackey-Glass attractors, we got ten different networks ($N = 1000$, $g = 1.5$), of both SARN and static type. For each of these ten networks, λ was numerically estimated (Fig. 3.6) for the spontaneous (no external input) network dynamics, as well as for the network trajectories elicited by each of the four contextual inputs (recall that

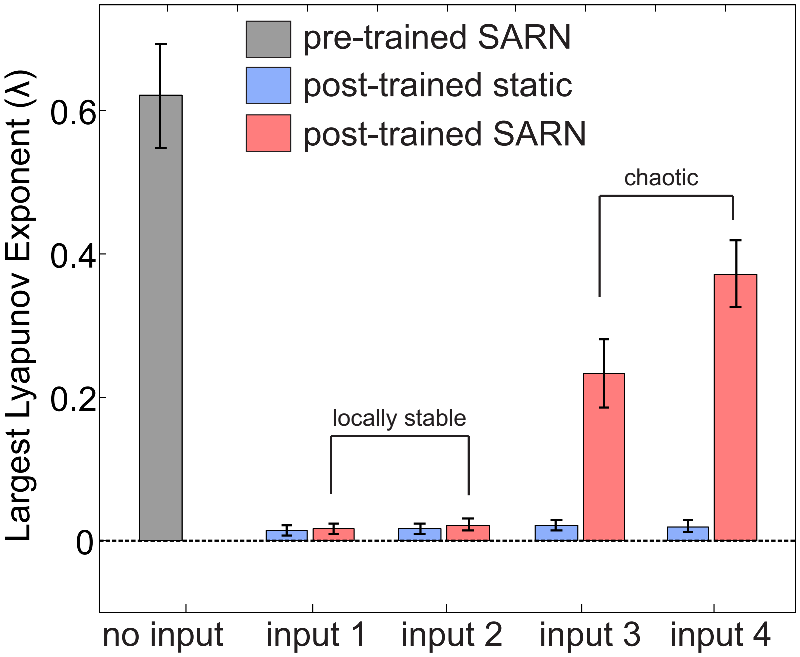


Figure 3.6: **Largest Lyapunov exponent estimation before and after training for SARN compared with static reservoirs** Plot of mean LLE values as estimated across ten trials for the ten different reservoir networks. The pre-trained SARN and static reservoirs both have high positive λ value 0.6213 ± 0.0727 in the no input condition, displaying chaotic spontaneous network activity (here only SARN results is displayed as both networks are essentially the same in the initialized state with $N = 1000$ and $g = 1.5$). Post training, all the four context inputs induced locally stable trajectories in the static reservoir network, indicated by the positive λ values very close to zero (input 1: 0.0138 ± 0.0064, input 2: 0.0155 ± 0.0073, input 3: 0.0204 ± 0.0077, input 4: 0.0188 ± 0.0089). As such the network dynamics were constrained in periodic stable activity, leading to the relatively good performance for generating stable Mackey-glass attractors, but very poor performance for the two chaotic attractors (see Fig. 3.5 (a) and (b)). In the post-trained SARN, context input 1 and input 2 reasult in very small positive, close to zero λ value, indicative of the induced network trajectories being locally stable ($\text{input1} : 0.0160 \pm 0.0072$, input 2: 0.0214 ± 0.0077). However in this condition, input3 and input 4 lead to diverging network trajectories as indicated by the significantly positive values of λ (input 3: 0.2331 ± 0.0469, input 4: 0.3723 ± 0.0467). As a result post training, the SARN network can be seen to encode both locally stable and diverging (chaotic) trajectories in an input dependent manner. Here the error bars indicate standard deviation of the estimated LLE values with 95% level of significance.

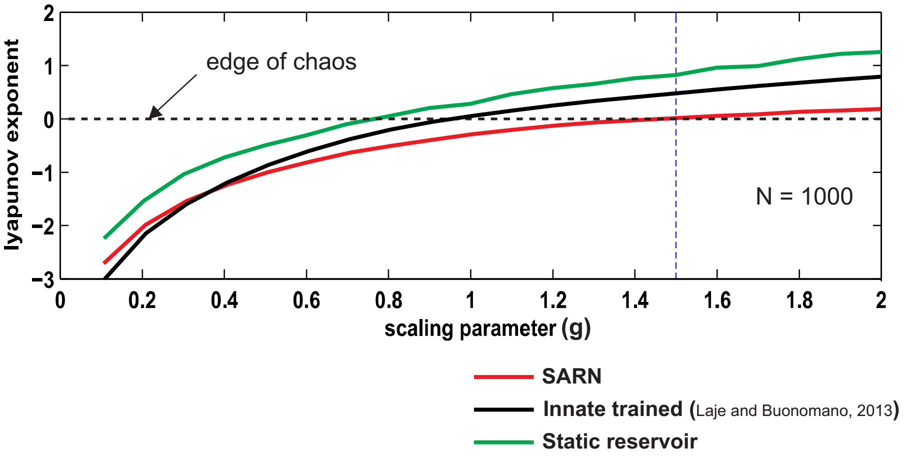


Figure 3.7: **Local Laypunov exponent of different reservoir networks, ploted against increasing scaling parameter value post training or local adaptation.** Here the three reservoir networks were randomly initialized with identical paramter settings. In case of the static reservoir (green line), LLE was estimated for the spontaneous activity of the originally created network. In case of innate trained (black line), after random initialization with each scaling value g, supervised innate learning of recurrent connections were carried out and LLE was estimated on the resultant trained network. For SARN, after each initialization, IP and timescale adpatations were carried out. LLE was then estimated on the adapted network. The blue dashed line shows the optimal value of $g = 1.5$ which resulted in SARN being at a critical regime.

these inputs were 1-of-4 encoded such that, at a time, only one of the channels were active and provided a brief 100 ms pulse signal). Prior to training, the reservoir networks initialized with a scaling parameter of $g = 1.5$, demonstrated exponentially diverging trajectories (chaotic dynamics) for the spontaneous activity indicated by the high positive mean LLE value. Both SARN and static networks showed similar chaotic dynamics owing to the scaling factor g being greater than one (Sompolinsky et al., 1988). However after training the network, the mean λ across the static networks for all the four context inputs was a small positive value (not significantly larger than zero), indicating locally stable dynamics. This clearly explains the reason the static networks, although learned to generate the periodic Mackey-Glass patterns, in the same network the other two inputs (with larger τ_m) failed to drive the readout neurons to generate chaotic patterns. On the contrary, in case of the SARN networks, context input 1 and input 2 (mapping to stable output patterns) result in a significantly positive mean value of λ, suggesting chaotic trajectories. As a result, SARN was able to learn to generate both the mildly and highly chaotic Mackey-Glass patterns. These results show that, the adaptation via intrinsic plasticity and neuronal timescales in SARN allows the network dynamics to be modulated differently by the context inputs, allowing both stable as well as chaotic dynamics.

Recently in (Laje and Buonomano, 2013), a new supervised synaptic plasticity rule called *'innate training'* for the reservoir recurrent connections (error gradient decent learning) was introduced, that essentially uses the spontaneous activity (innate non-noisy) of the reservoir neurons as the target signal to adapt the recurrent weights in the presence of internal noise and external inputs. They showed that using this method it was possible to enable the network dynamics to exhibit both chaotic as well as locally stable trajectories. However our results here clearly demonstrate that, even in the absence of such biologically implausible innate supervised plasticity, local adaptations of neuron firing rate curve (IP) and neuronal time constants, in combination with standard supervised synaptic plasticity (where the target signal is the desired output and not an innate trajectory) was sufficient to demonstrate both stable and chaotic network dynamics, which also significantly outperformed static reservoir networks. In order to quantify further, the impact of our local adaptation mechanisms on the network dynamics, we estimated the local Lyapunov exponents for three similarly initialized, same size (1000 neurons) reservoir networks, a static version, an innate trained version (Laje and Buonomano, 2013) and SARN. In all cases we varied the scaling parameter g ($g \ll 1$ - spontaneously sub-critical to $g \gg 1$ spontaneously chaotic) and estimated Lyapunov exponents after training or local adaptation of the networks. As observed in Fig. 3.7, with increasing values of the scaling parameter, the estimated Lyapunov exponent also increases, indicative of gradually diverging dynamics. However, the largest increase in the estimated Lypaunov exponent was observed for the static reservoir network, suggesting strongly chaotic network dynamics when $g > 1.0$ (it should be noted that this is the spontaneous network activity). In comparison the innate trained reservoir network demonstrated critical dynamics at $g = 1$ ($\lambda = 0$) and then slowly increasing λ for $g > 1$. As a result the overall network still remains locally stable or near critical for g close to one, enabling the network to still perform well under the influence of large noise. However, in case of SARN, after adaptation, larger values of the scaling parameter did not have a strong impact on the network dynamics with the estimated λ value being very close to zero. This indicates that the overall network dynamics largely remains close to the critical point or the so called edge of chaos with $\lambda \approx 0$, leading to the optimal levels of computation (Legenstein and Maass, 2007a) and information storage and transfer (Boedecker et al., 2012) in SARN. As such the above results clearly demonstrate that local adaptations in the reservoir network not only enable input dependent modulation of network trajectories in different dynamic regimes, but overall, the spontaneous dynamics of the adaptive plastic network is inherently closer to the critical limit (optimal scaling parameter in the above example is indicated by the blue dashed line in Fig. 3.7, for $g = 1.5$).

3.1.4 Intermediate Summary

Using complex time-series processing tasks as abstraction of biological temporal information processing (in the short timescale of milliseconds to seconds), in this section we clearly demonstrated that with local unsupervised adaptation and plasticity, SARN significantly outperforms its static counterparts. It not only enables robust non-linear computations but can also achieve extended memory capacity for a similar sized network (without the need for specialized network topology) (Boedecker et al., 2009). Real environmental time-varying stimuli is mostly composed of multiple timescales. Often with fast and slowly varying components. Using the periodic to chaotic fluctuations of the Santa-Fe real laser data, we showed that adaptation of individual neuronal time constants can indeed enable reservoir networks to effectively deal with such multiple timescales or multiple frequency components of incoming signals. Although it has been traditionally difficult to train a single RNN to generate chaotic patterns for arbitrary periods of length (Jaeger and Haas, 2004), here we showed for the first time, that with our adaptation mechanisms SARN can learn to represent and generate both stable and chaotic attractors in the same network. Furthermore, Lyapunov exponent analysis proved that post training, SARN operates at the near critical regime or the edge of chaos, which has been shown to be optimal for computation. It should be noted that this is in contrast to other existing self-organized RNN models (Lazar et al., 2009), where in a combination of synaptic and homeostatic plasticity mechanisms has been shown to lead to sub-critical dynamics. In the next sections we dive into the workings of SARN even further from the perceptive of behaviorally more relevant temporal processing scenarios in the timescale of seconds to minutes, like delay temporal memory, time perception and complex motor processing.

3.2 Timing and Delay Temporal Memory

The ability to tell time is critical for the learning of ordered motor behaviors as well as the underlying cognitive processes, in all living creatures. However the mechanisms by which biological brains tell time is still largely unknown. As such, in this section, initially, we show that the underlying dynamics of SARN can be used naturally to generate clock like, timed responses. Furthermore, it also captures the experimentally observed variance signature of timed responses, that typically follow generalized Weber's law where the variance of specific responses are linearly related to the square of the interval being timed (Buhusi and Meck, 2005). Unlike previous models of RNNs, that have used specific mechanisms of short-term synaptic plasticity (Buonomano, 2000) or supervised recurrent layer noise suppression techniques (Laje and Buonomano, 2013), in order to discriminate temporal stimuli, here we show that the local adaptation mechanisms of SARN were sufficient to learn to respond at specific intervals of time while also capturing the underlying Weber's relationship. Finally, we demonstrate the ability of the same network to robustly perform in delayed response tasks, that require temporary storage of time varying input signals in order to make future decisions. This is applied in a closed loop embodied bio-inspired walking robotic system thus highlighting the functional relevance of our model to 'understanding the brain by creating the brain' (Kawato, 2008) approaches.

3.2.1 Learning to Tell Time: Responding at Specific Intervals

In order to characterize the ability of SARN to generate responses at specific intervals of time we use the same general experimental setup as in the previous section 3.1, using a 1000 neuron recurrent network, scaled with a factor of $g = 1.5$. In this case only a single input neuron was connected to the recurrent layer that provided a brief 50 ms square pulse of high amplitude (2.5) as input to the reservoir network. A single readout neuron was connected with the objective of generating pulse output signal at different time intervals (100 ms to 1 sec). The desired output signal was flat at all times with a small finite value except at a precise time intervals when it pulses. The recurrent network state activity was governed as before by the standard equation 2.14. In order to introduce additional variability and test the robustness of the network, Gaussian white noise with zero mean and standard deviation 0.001 was introduced inside the network as the bias term B_i for each recurrent neuron. 20 test trials were carried out after adaptation and supervised training of the network, in order to produce timed responses at increasing intervals of time starting from 100 ms.

As observed in Fig. 3.8 (a), a brief input pulse of 50 ms was used to stimulate the reservoir network, transiently. After this initial burst of input, the network receives no further external stimuli. In this example, the network was required to pulse exactly after a 1 s or 1000 ms time interval (given by the green signal). Since this is a continuous system, in between the initial pulse and the final desired pulse, the network goes into spontaneous dynamics, as indicated by the activity of few randomly selected recurrent layer neurons. Here we plot the signals from multiple trials. Pre-training of the network to fix the reservoir neuron non-linearity and time constant parameters were carried out initially using 50 epochs of noisy (standard deviation 0.01) versions of the input pulse and varying the pulse duration between $50 - 150$ ms. As indicated by the middle plot in Fig. 3.8 (a), after training SARN, prior to any input pulse coming into the reservoir, there is considerable trial-to-trial variability in the dynamics of the network, with diverging trajectories. However, post training, on the arrival of the brief input pulse, the individual recurrent neurons follow a single trajectory (still diverse between neurons owing to the recurrency and different time constants in the network) with negligible variation in between trials. At the same time, during the period when there is no input to the network, the readout neurons activity is also considerably noisy and varies in between trials (as indicated by the red signal in Fig. 3.8 (a) below). Post training, however, the readout neuron was able to successfully learns to remain quiescent until the training delay period of 1 s and then produces an output pulse, almost perfectly matching the desired signal. Unlike the previous examples using the benchmark data-sets, in this case, since the input to the network is very brief in duration, in order to generate outputs at significantly longer time intervals, it needs to effectively make use of fading memory of incoming input stimuli. As such, here we see that SARN after training, clearly learns to behave like a clock model, with the precisely timed output pulses reminiscent of ticks at specific intervals by a mechanistic clock. Furthermore the stability of the clock like behavior can be observed by taking the norm of the reservoir-to-readout synaptic strengths ($|W^{out}|$). If $|W^{out}|$ is large in value, then even small disturbances in the internal recurrent neural activity will be greatly amplified in the readout activity leading to unstable behavior. As observed in Fig. 3.8 (b), after training $|W^{out}|$ remains considerably small and without fluctuations after the input pulse arrives into the network, resulting in stable output in the readout neurons at the exact time interval.

We repeated the same procedure across 20 trials, however with a much larger noise (zero mean with standard deviation 0.1), and for different time intervals, ranging from 100 ms to 1000 ms. The large noise was used deliberately in order to quantify the variance (in milliseconds) of the peak of SARN generated signal (timed pulse output) and the desired time interval of the pulse. Experimental studies have shown that in a given task (involving timed responses), the variability of timed responses are often well described by Weber's law, i.e. there is a constant ratio between the standard deviation of the response and the interval being timed (Buhusi and Meck, 2005). Specifically for motor timing involving temporal processing or timing in the timescale of a few milliseconds to seconds, it has been established that the variability is well described

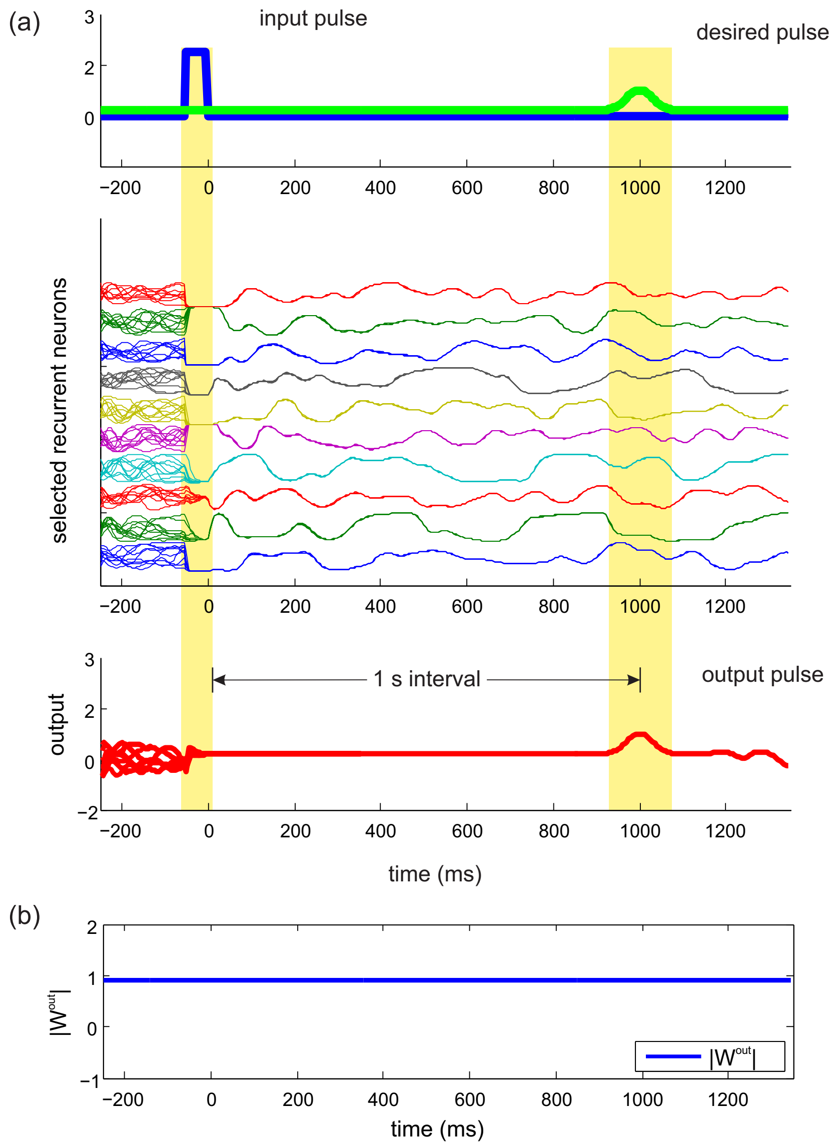


Figure 3.8: **Clock like behavior with SARN, learning to generate pulses at specific intervals of time** (a) A brief 50 ms pulse is provided as driving input to the reservoir network (blue). The network is expected to learn to generate precisely timed response after 1000 ms, given by a desired Gaussian pulse (in green). In between, the network recieves no further external inputs. The middle plot shows activity of select reservoir neurons after adaptation and supervised learning, across 20 trials. Prior to the arrival of the input pulse, network activity is highly irregular with each neuron following randomly different trajectories in between trials. After learning however there is little trial-to-trial variance in the trajectories of the individual neurons. The bottom plot shows the learned output from the reservoir network across all trials. The readout neuron activity is random prior to the arrival of the input pulse, but after presentation of the input, it pulses precisely after a dely of 1000 ms or 1 s. (b) The plot of the norm of reservoir-to-readout weights W^{out} showing a small steady value after learning. As a result the output remains stable even in the presence of high activation values of reservoir neurons. Trials were carried out in the presence of Gaussian noise with zero mean and standard deviation 0.01.

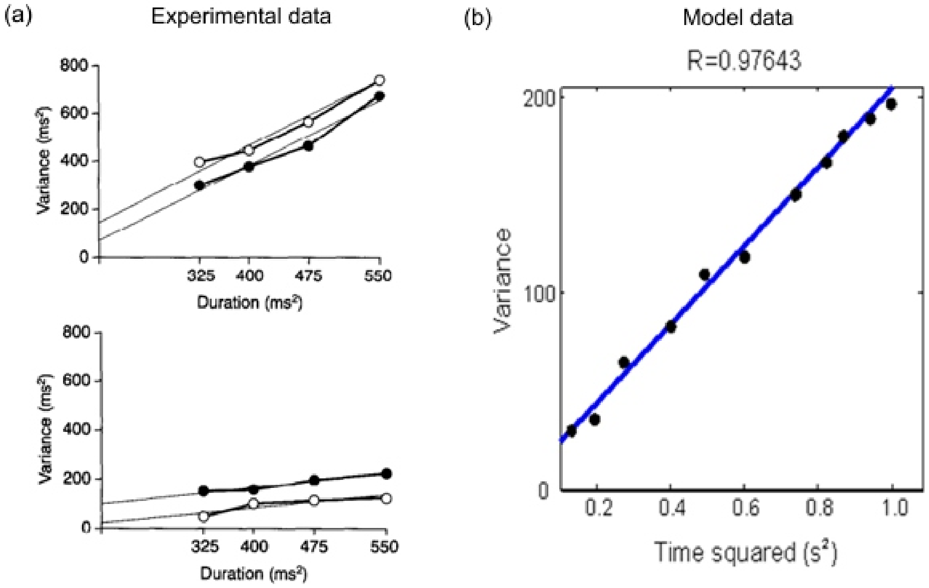


Figure 3.9: **SARN matches experimentaly observed generalized Weber's law of a liniear relationship between peak time variance and square of the timed interval** (a) Experimental data obtained from time perception (●) and production tasks (○) from (Ivry and Hazeltine, 1995) showing a linear relationship between the mean variances (σ^2) and the square of the duration timed (s^2), capturing the generalized Weber's law where the liniear relationsip follows, $\sigma^2 = k^2 s^2 + C$. Where k is the slope of the line and C is the intercept on the y-axis representing a time independent component of the variance. (b) SARN reproduces similar liniear relationship (high regression coefficent of $R \approx 0.98$) between square of the different timed intervals from 0.1s to 1s and their mean variances. It was also able to capture the underlying generalised Weber's law with a non-zero Y-intercept demonstrating that it indeed acts as a model for biological time perception within the timescale of milliseconds to seconds.

by the generalized Weber's law (GWL). Wherein, the variance of the response is linearly related to the square of the interval being timed, plus an additional variance term which is time duration independent (Ivry and Hazeltine, 1995),(Hazeltine et al., 1997). This experimentally observed variance signature of timed responses has thus been used as an important criterion to evaluate different models of timing (Laje and Buonomano, 2013). Here, our SARN model when tested with the different timed intervals, also captures well this underlying linear relation of the generalized Weber's law of timing. In figure 3.9 left, we plot the original experimental results showing this linear relationship as obtained from two different interval timing tasks, namely time production and time perception (Ivry and Hazeltine, 1995). As observed in figure 3.9 right, with a noise level of 0.1 at each reservoir neuron, similar to the experimental results, we obtained a linear fit (with a regression coeffecient of $r > 0.95$) for the square of the time responses with the variance of the time of peak of the output signal. Furthermore, as expected from the GWL, the slop did not intercept at zero, but at a finite positive value of variance, which provides in this case the additional time independent term of the relationship. However, in this case changing the amount of noise or training on very large time intervals (affecting the total timing capacity) may lead to a non-linear relationship. Nevertheless, the adaptive mechanisms in SARN clearly enable it to behave as a timing model (atleast in the short timescale of milliseconds to seconds), that also fulfills the critical criteria of a linear increase in temporal variability with increase in interval duration (squared) as observed in time perception by biological systems.

3.2.2 Delay Temporal Memory with Artificial Agents

In order to use a brief input stimuli to produced precisely times responses at specific intervals, the reservoir network needs the ability to transiently hold a fading memory of the incoming time-varying inputs. Thus a natural extension of such a clock like or timing mechanism is towards delay temporal memory to guide specific behaviors of organisms. We use the term delay temporal memory to describe the temporary storage of input or driving stimuli to the network for finite delay periods. Here we demonstrate the temporal memory capacity of our system, by employing a behaviorally relevant, variable delay temporal memory task of navigation through a T-shaped maze. The experiments are carried out using a complex physical walking robot AMOS-II Fig. 3.10. For this task we used a moderate reservoir size of $N = 500$ neurons. This was fixed keeping in mind the extended delay memory required for the T-maze in the real robot experiments. All other default parameters were initialized with values similar to the experimental setup in the previous sections. The network consisted of 4 input neurons providing sensory time-varying information to the reservoir and 2 readout neurons that controlled the left and right turn motor neuron activity of the robot.

The primary objective of this task is to let the robot move from the starting position until the end of the maze while making the correct turn at a recall zone (see Fig. 3.10 (c) and Fig. 3.11). While walking along the corridor, the robot receives a brief *cue* signal (a bright light activation for a short duration of time) either to their left or right side. This provides information to the robot regarding the required turning behavior at the T-junction. On reaching the end of the corridor the robot should make the correct turn corresponding to the side at which the signal

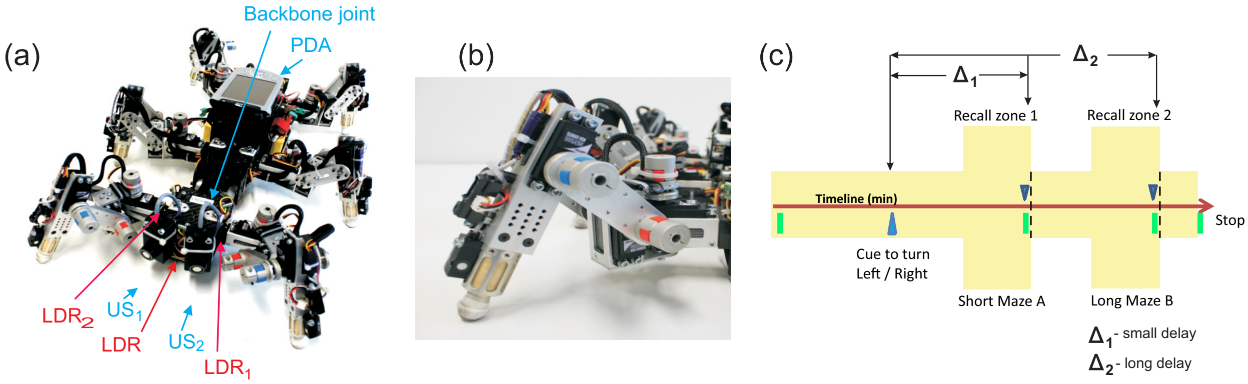


Figure 3.10: (a) Biologically inspired six-legged walking machine AMOS II. (b) Leg structure of AMOS II inspired from a cockroach leg (showing the three different leg joints). (c) The delay temporal memory T-maze navigation setup showing short (Δ_1) and long time delays (Δ_2) for maze A and B, respectively. Cue is given as a light signal either to the right or left of the robot, indicating the corresponding turn at the recall zone (T-junction).

was given.

In order to demonstrate the generalization capability of the system to longer time delays, we divided the task into two mazes (see Fig. 3.10 (c)) of different lengths. Maze B requires a significantly longer temporal memory (larger delay between cue and recall) as compared to maze A. This delay period is typically in the timescale of a few minutes. Here the robot had to learn both the reactive behavioral task of turning at the T-junction as well as recall the cue signal shown transiently at a much earlier point in time, to negotiate the correct turn. Specifically, here one can imagine the cue signal providing the correct context information to the network based on which one of the readout neurons need to be active at a precise time interval (length of maze). Hence, in the absence of the recall signal, this is akin to the previous task of generating clock like behaviors. The recall signal at the T-junction provides the necessary correlation signal to decide when to turn, while the previous cue signal gave the information about in which direction to turn. As such, here we can avoid the use of conventional methods like using landmarks to identify the T-junction.

Description of the complex walking robot AMOS-II

AMOSII (successor to AMOS robot (Steingrube et al., 2010)) is a biologically inspired hardware platform (Fig. 3.10) consisting of six identical legs. Each leg has three joints. The morphology of these multi-jointed legs is modeled on the basis of a cockroach leg but with the tarsus segments ignored. The body of AMOS II consists of two segments: a front segment where two forelegs are installed and a central body segment where the two middle and the two hind legs are attached. They are connected by one active backbone joint inspired by the invertebrate morphology of the American cockroach's trunk. This backbone joint is for up- and downward bending, which allows it to climb over obstacles. All leg joints including the backbone joint are driven by digital servomotors.

The size of AMOS II is 30 cm wide, 40 cm long, 22 cm high. The weight of the fully equipped robot (including 19 servomotors, all electronic components, sensors, and a mobile processor) is approximately 4.5 kg. AMOSII has a total of 17 sensors. For the maze-navigation experiments we only make use of the two light dependent resistor sensor ($LDR_{1,2}$) on the left and right sides of the front body part, and the front two ultrasonic sensors ($US_{1,2}$). These act as the sensory inputs to the reservoir network for the T-maze navigation task. We use a Multi-Servo IO-Board (MBoard) installed inside the body to digitize all sensory input signals and to generate a pulse-width-modulated signal to control servomotor position. The MBoard is connected to a personal computer (PC) via an RS232 interface. Electrical power supply is provided by batteries: one 11.1 V lithium polymer 2,200 mAh for all servomotors, two 7.4 V lithium polymer for the electronic board (MBoard) and for all sensors. For more information of AMOSII, please refer to Ren et al. (2012) and Manoonpong et al. (2013b).

The experiment consisted of 3 parts. In the first part data-set acquisition was done using human controlled navigation of AMOS-II through the maze and the sensor and steering signal (see Figs. 3.11 (a) and (b)) readings were recorded. These act as the inputs and desired outputs of the network respectively. 20 runs with different starting positions and for both left and right turn cue were carried out. This was done for both small and long time delays between cue and recall zone. This data was then used for the supervised learning and pre-training via IP and AIS based time constant adaptation of the reservoir network. Finally online testing was carried out with the trained steering signals being fed into the AMOS-II controller.

In this setup the longest maze B, had a delay time fifteen times larger than the longest interval (1000ms) presented in the previous clock example, of the order of 1500 time steps (1 time step $\approx$ 100ms) between the cue and the recall (Fig. 3.11 (a)). Here, AMOS-II locomotion was driven by modular neural locomotion control mechanism (see Fig. A.1 in appendix A.3) (Manoonpong et al., 2013b), with the learned output from the reservoir network being used to steer the robot in the left or right direction (it controls the VRN network in Fig. A.1). In Fig. 3.11 (a), we plot the sensor signals, that act as the time-varying input stimuli to the reservoir. The onset of LDR_1 triggers the left turn cue, while the simultaneous onset of both the front ultrasonic sensors $US_{1,2}$ signals at the recall zone. A high-dimensional (on the 500 dimensional reservoir state space) convolution of these signals reverberate as neural traces inside the reservoir network (a subset of these diverse set of signals is plotted in Fig. 3.12 (b)). The local active information storage (Eq. 2.17) used to modulate the time constants of individual neurons in SARN (Fig. 3.12 (a)) shows that the time of the two events of *cue* and *recall* are recognized as high information content regions (500 time steps and 1500 time steps, respectively), while the reservoir neurons have a relatively low local AIS value during the remaining time steps. As a result, due to the mechanisms of timescale adaptation based on Eq. 2.19, the neuronal time constant or decay rate τ_i gets modulated such that most neurons have a low decay rate (high local memory) at the time of the left or right turn cue and then again at the end of the corridor (Fig. 3.11 (c)) when recall signal gets triggered. During the remaining time steps, the reservoir neurons have a higher decay rate (low local memory). As local neuronal decay rates or time constants act as their local timescales and collectively control the timescale of the reservoir network (see Eq. 2.14); this mechanism leads to a slowing down of the reservoir dynamics at high information content regions (information storage) of cue and recall, and speeding up during the rest of the time.

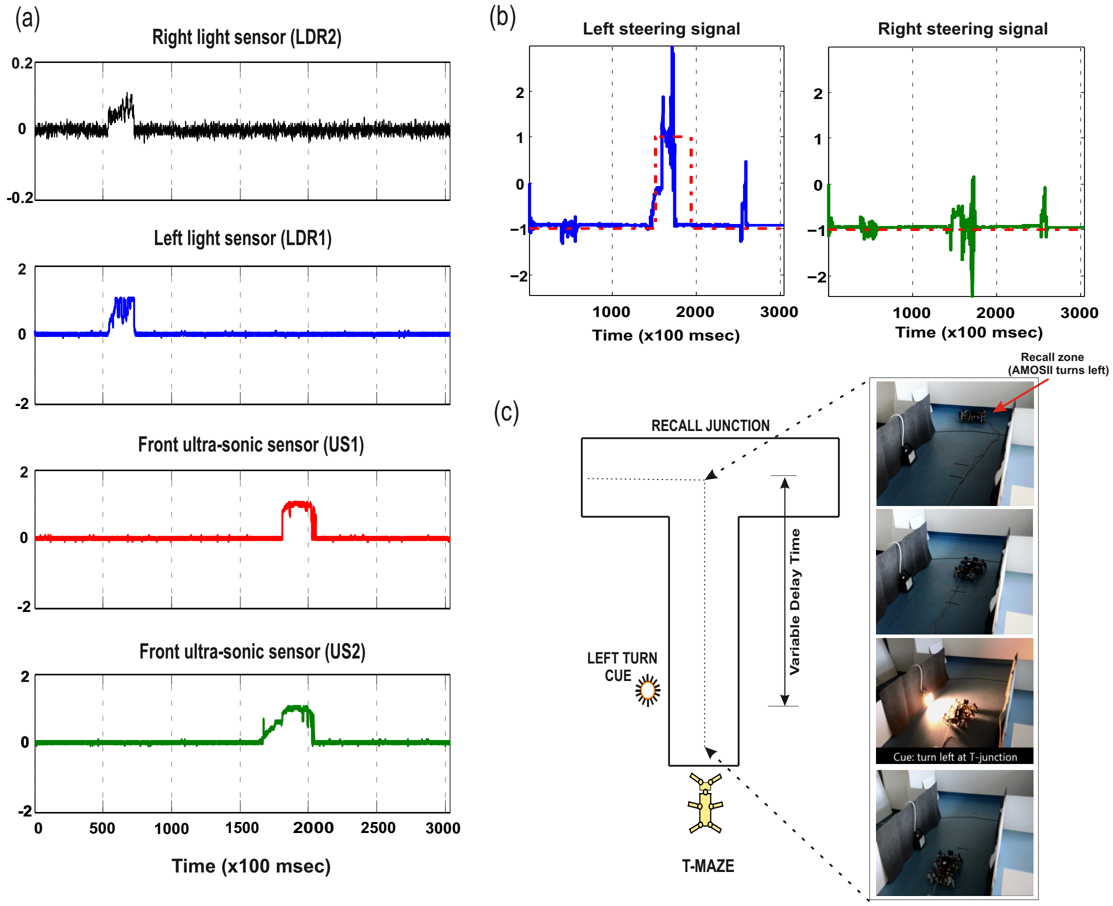


Figure 3.11: (a) Plots of the sensor signals from AMOSII recorded during the experiment, which act as the four inputs to the reservoir network. The signals shown are from a single run where the cue signal (light source) was applied to the left, while walking along the corridor (LDR1 $\gg$ LDR2). The two ultrasonic sensors become active at the same time when AMOS-II reaches the T-junction (cue recall zone). (b) Plots the trained reservoir network outputs (Solid-line: learned behavior; Dotted-line: desired behavior). Here the left steering signal is active (+1) while the right steering signal is inactive (-1) and the robot makes a left turn (behavior learned at the same time step of the activation of the US1 and US2 sensors indicating the recall zone. (c) Pictorial representation of the T-shaped maze setup. While walking along the long corridor, a cue in the form of a light signal is applied either to the left or right side of AMOSII. The robot needs to recall this cue at the *recall junction* and execute the corresponding turning behavior. The temporal delay between the time of presentation of *cue* and the end of the corridor (T-junction) is the total memory span. This can vary with different delay times for small and long mazes. The screenshots (right) from the experiment show the actual behavior of the hexapod while walking along the corridor.

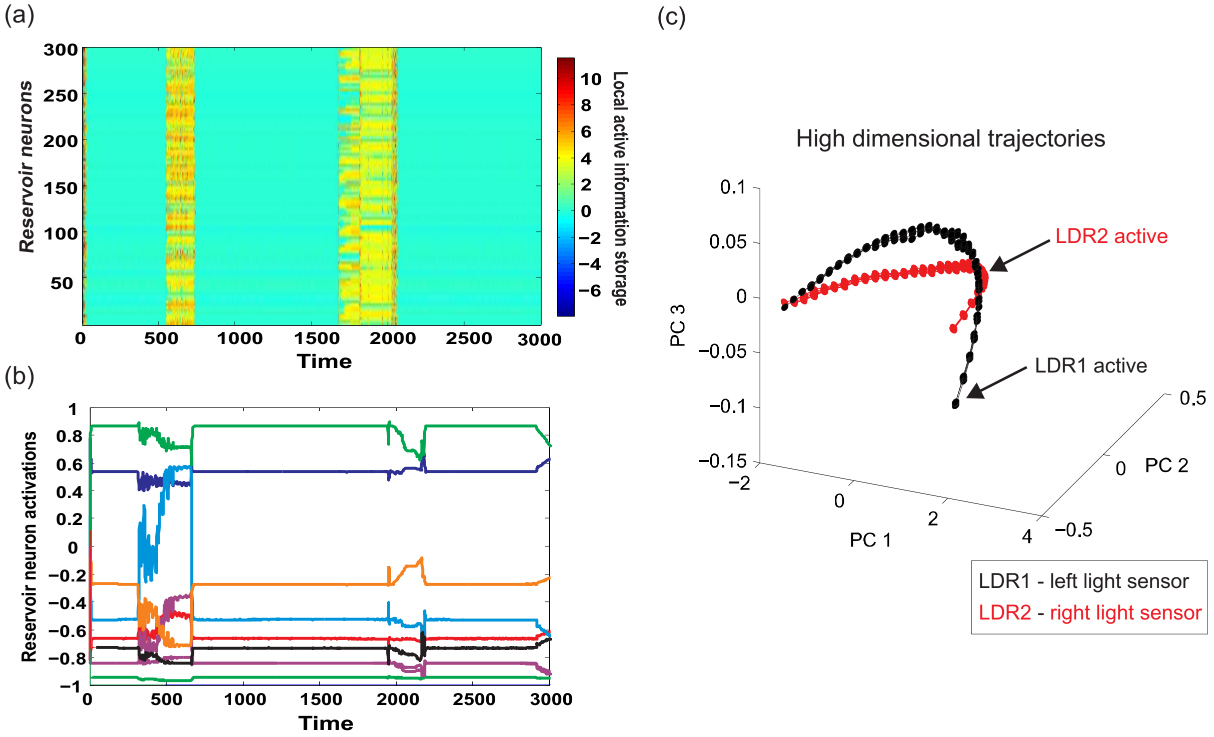


Figure 3.12: (a) Plot of the Local active information storage values for a subset of 300 reservoir neurons at different time steps (color coding corresponds to the local active information storage values at different time steps). (b) Reservoir activations for a randomly selected subset of the neurons. (c)Projection of the network activity on the first three principle componets. Depending on the input signal the network follows distinct and separate trajectories through the high-dimensional state space. These trajectories can be imagined as stable dynamic attractors in the network. As such the readout neuron is able to discriminate between a left and right turn by following a path along one of these trajectories. Here each dot represents the value obtained from a given trial for a single maze, at each point in time.

Using the online supervised learning mechanisms, the reservoir network successfully learns the correct turning behavior. In this case due to the previously applied left turn cue, only the left steering signal is active, while the right steering signal remains dormant and the robot makes the corresponding turn. It is important that the robot starts turning at the correct time in order to prevent an early turn or crashing into the wall of the corridor. This was clearly achieved as seen from the near perfect coincidence between the desired and the actual output signals (Fig. 3.11 (b)).

The reservoir outputs were post-processed to get rid of signal noise before being feed into the modular neural controller of the robot. The robot was tested on both short maze A (short delay time) and long maze B (significantly longer delay time) setups and the learning performance (measured as percentage of successful turns) was averaged over 20 runs for both left and right turn scenarios. As demonstrated in Fig 3.13 (a) and (b), in case of the shorter maze, SARN

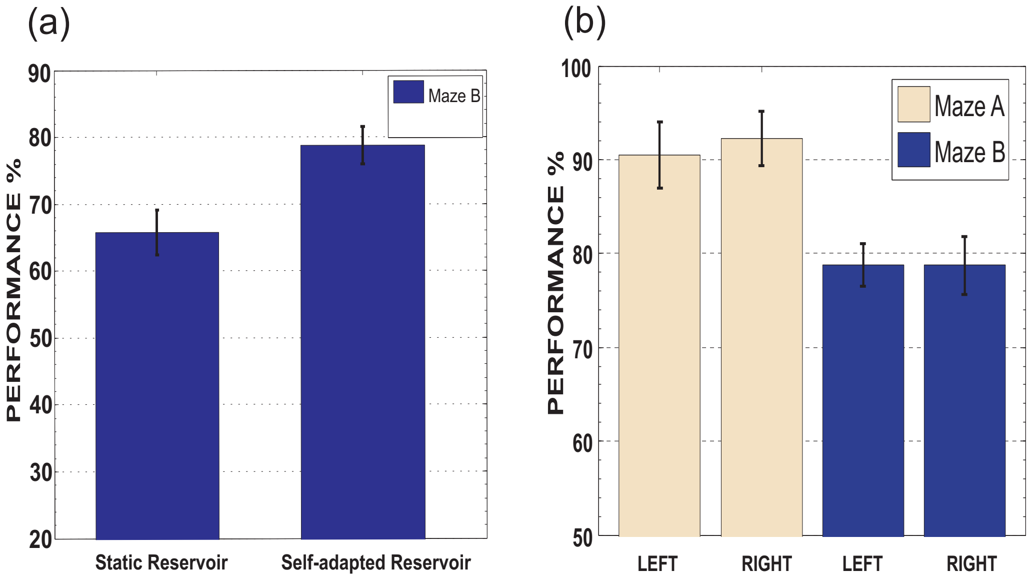


Figure 3.13: ((a) Performance on the large maze B task after 20 trials for static reservoir vs our self-adaptive reservoir. Our network outperforms by an order of 10%. (b) Performance of the robot in both mazes (Maze A shorter than Maze B) measured in terms of the percentage of correct times the robot took the proper trajectory (Left/right turn at the T-junction) to reach the end point. 5% noise is considered on all sensors.

achieved a performance of 92.25% (±2.88 standard deviation). A good generalization capability for the longer maze B was also observed with the average performance of 78.75% (±3.11 standard deviation), both for right turn. This was significantly larger than the performance obtained for a static reservoir network (Antonelo et al., 2008) for a similar task. Without timescale adaptation, in case of the static reservoir, AMOSII showed a wall following behavior with turning being triggered prematurely and the output signals reconstructed without threshold crossing (less than 1). It should be noted that in the absence of the recall signal, the task was similar to the clock example, however with an extended delay period. Thus here we clearly show, that not only SARN can generate timed responses, but can also produce robust short-term (temporal) memory guided behaviors in complex artificial systems, while also outperforming its static reservoir counterparts. The overall performance of SARN could be further enhanced if additional sensors were made available to the robot, owing to the availability of additional discriminatory input signals to the reservoir.

The effect of the time varying input signals to the reservoir dynamics was further accessed using principle component analysis (PCA) of the reservoir network activity. Fig. 3.12 (c), plots the projection of the 500 dimensional network activity onto the first three principle components (here the first five principle components explained 98% of the variance in the activity) in order to visualize the actual input dependent trajectory the network follows. Each point shows the result of 10 different runs to calculate the PC's. Depending on the input signal that is active (LDR_1 for left light sensor and LDR_2 for right light sensor) the network dynamics follows two separate trajectories through the high dimensional space, which is robust across the different trials. These trajectories act as dynamics attractors, that given the current inputs constrain the network activity to follow a stable direction along its path. As a result, the the readout neurons robustly learn the corresponding behavior of turning left or right. This in combination with the previous results of timing behavior, proves that SARN can not only hold time-varying inputs transiently in its activity to produce outputs at particular points in time, it uses inputs as contextual information to learn separate high dimensional attractors (trajectories) through its state space. Recent experimental studies have also demonstrated such context-dependent processing by the recurrent dynamics in the pre-frontal cortex (Mante et al., 2013) In the next section we will see how such context-dependent information and the network transient dynamics can generate complex motor patterns which is also robust to external perturbations.

3.3 Complex Motor Pattern Processing

The generation and processing of complex motor patterns forms one of the essential outcomes of robust temporal information processing in the brain, within the timescale of few milliseconds to seconds. A number of recent experimental (Churchland et al., 2010) and theoretical studies (Hennequin et al., 2014), have shown that the execution of limb movements involves complex transient dynamics within populations of neurons in the motor cortex. As such, here we show that our adaptive and plasticity mechanisms in SARN complements the inherent transient dynamics of the network, by successfully learning complex time dependent motor behaviors. This is presented as a natural extension of the previous section, where in, we teach the reservoir network to generate different handwritten patterns using high-dimensional temporary input stimuli and contextual inputs, which is also stable to external perturbations or noise.

Specifically here we create an interval timing dependent handwriting generation task for a multiple joint robotic arm. A 3000 neuron SARN model was used in this case (Fig. 3.14 (a)). All other parameters were initialized similar to the experimental setup in section 3.1.1. Handwriting data for all the 26 letters of the English alphabet were collected using a human participant[1]. Each letter in this case was represented by two time-varying signals that maps the letter onto a 2-dimensional (x and y co-ordinates) surface. These provide 26 × 2 dimensional inputs to the network, and was presented as a brief stimuli of 210ms duration (Fig 3.14 (b)). The network also received a fixed auxiliary bias of 0.8 as a constant input. Additional two context inputs were given as a brief 100ms square pulse starting at 250 ms after the network was initialized. This was encoded in a 1-of-2 scheme, such that at a time only one of the context signals was active and the other remained zero. Two readout neurons were connected to the reservoir which were trained to generate the x and y coordinate values (2-dimensional time-varying signal) of the words '*memory*' and '*apple*', after a delay period of 150 ms from the time the context signal ends (see color coding in Fig. 3.15 (a)). The learned x and y coordinates were then transformed into joint angles of a multiple joint robotic arm using inverse kinematics, such that it learns

---

[1] Handwriting samples were taken from a single person, where the person was asked to write single letters ('a' to 'z'). Data were obtained by the use of a pen tablet (Wacom Intuos3 A3 Wide DTP) with a size of 48.8 cm ÃŮ 30.5 cm, resolution of 5080 lpi and a sampling rate of 200 Hz.

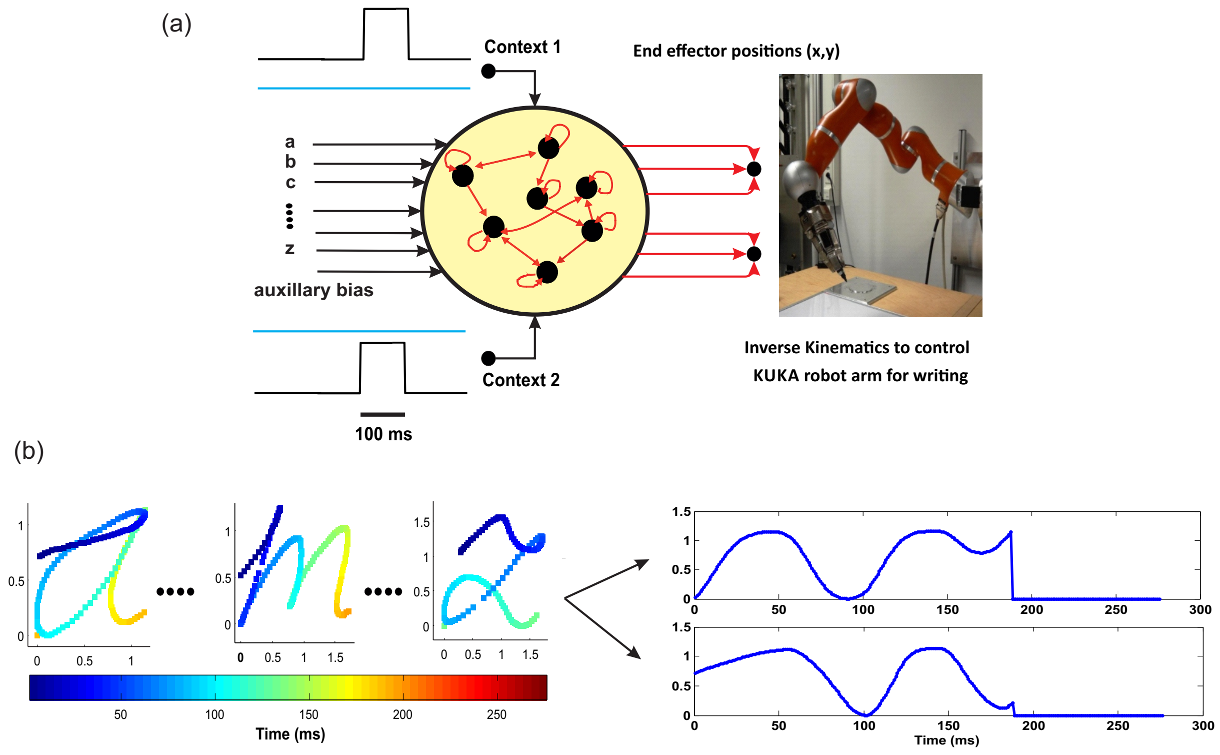

(a)
Context 1
End effector positions (x,y)
a
b
c
z
auxillary bias
Context 2
100 ms
Inverse Kinematics to control
KUKA robot arm for writing
(b)
Time (ms)
Time (ms)

Figure 3.14: **Time interval based complex motor pattern generation with SARN**(a) A 3000 rate coded neuron SARN used for the complex motor pattern generation task. Inputs consisted of 26 × 2 time series data of handwritten English alphabets, along with two context input signals encoded in a 1-of-2 encoding scheme (one one active at a time). The active signal was presented as a brief stimulus of 100 ms duration starting at 250 ms time point after the network was initialized. The network also received a fixed auxiliary bias of amplitude 0.8. The readout layer consists of two neurons encoding the x and y trajectories for right either the word 'memory' - context 1 or the word 'apple' - context 2. These x and y values were converted into joint locations for a multiple joint KUKA robotic arm using inverse kinematics (right figure). (b) The input trajectories for alphabets 'a', 'm' and 'z' color coded by the time of of activations. The right panel shows the x and y time series data for the word 'a' that is the actual input to the reservoir. This was active for a maximum time interval of 210ms. After this period the network receives no time series input other than the constant auxillary bias signal. Only at 250 ms time point one of the context signal was activated for a brief duration signaling the readout neurons to learn to write one of the two words.

to write the corresponding word. The task was designed such that the context signal-1 should trigger the network to learn to generate the word 'memory' starting at the precise time point of 500ms, while context signal-2 should trigger the network to learn to generate the word 'apple' starting at the same precise point in time. Thus in order to successfully learn the task, the networks needed to perform both interval timing as well as learning the exact spatio-temporal pattern of activity (motor output) based on transiently active input signals.

After pre-training the network with IP and timescale adaptation based on the inputs, supervised learning on both the desired output trajectories (2D signal for each word), SARN was able to robustly learn to write both words, with remarkable accuracy starting at the precise time point of 500ms (see Fig. 3.15 (a)). PCA on the 3000 dimensional network state space showed that before learning and adaptation, the reservoir dynamics followed a particular trajectory through the high dimensional state space (Fig 3.15 (b) left). However after learning, the two context inputs were able to elicit two distinctly separate trajectories (Fig 3.15 (b) right) through the network space, thus enabling the readout neurons to generate the corresponding motor pattern or word. Furthermore, in SARN, these trajectories are locally stable and act as dynamic attractors, such that the network dynamics remain stable to external perturbations. This can be clearly seen by perturbing the network activity after the readout neurons have already started generating the desired trajectory. We perturbed the network using a 200ms pulse with considerably high amplitude of 0.5, of an additional input connected to all the neurons in the recurrent layer of the reservoir starting at 1500 ms time point (at the time of the letter 'e' in the word 'memory'). As observed in Fig. 3.16 (a) and (b), the external perturbation knocks the network out of its original trajectory, however within a few milliseconds the network was able to recover to its original trajectory (see 3.16 (b) zoomed plot) and continue generating the exact learned motor pattern. This demonstrates the ability of SARN to encode locally stable dynamic attractors as high dimensional trajectories through its network space.

Furthermore, clear separation of states or these trajectories can be observed for different context inputs. Thus this provides a crucial link to the experimental evidence for context dependent decision making in the cortex (Mante et al., 2013), as well as model for self-adaptive processes in RNNs (simple abstraction of the cortex) that by way of stable transient dynamics can sustain such complex motor behaviors. It should be noted that in a recent work from Laje and Buonomano (2013), a similar handwriting generation task was demonstrated (however with significantly less perturbation time 10ms and without a delayed input component), in order to motivate the existence of locally stable channels in otherwise chaotic RNNs. They however used a specifically created supervised learning rule that let reservoir neurons to learn their own innate trajectories in order to generate such complex behavior. Here, we demonstrate that even in the absence of any such specialized supervised mechanism, local adaptation of neuronal timescales coupled with biologically realistic intrinsic plasticity mechanisms are sufficient to generate complex motor patterns in a noise robust manner. We further quantified, the performance gained by our self-adaptive model by comparing it to the state of the art static chaotic RNN (Sussillo, 2014) and the 'innate trained' RNN model from Laje and Buonomano (2013). In all three cases, the same network size of 3000 neurons were used, with their individual parameters optimized for this particular task.

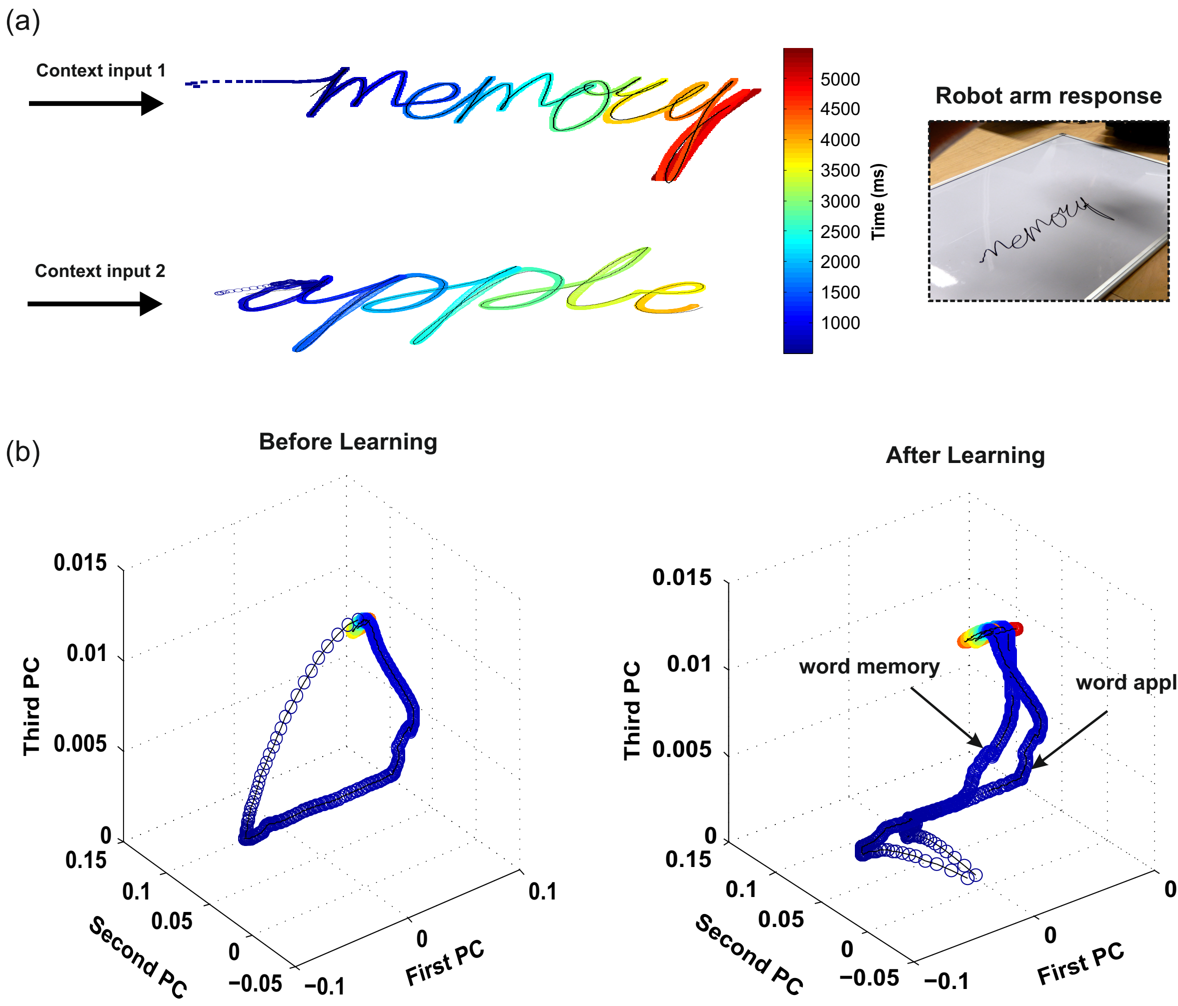


Figure 3.15: **SARN learns to generate the correct motor pattern depending on the context input**((a) Context dependent motor output generated by SARN color coded by the time of each event. The black line shows the actual desired trajectory for each pattern (words), (right) learned response of the KUKA robot arm for writing the word 'memory'(b) PCA on the reservoir network activity (left) before learning and adaptation there is a fixed trajectory through the high dimensional network space, (right) after learning and adaptation, different context inputs results in distinctly separate trajecories. These act as *dynamic attactors* such that the network dynamics start close to each other but follow different paths along each trajectory depending on the current conext input. All trajecories are color coded by the time of evolution as above. Separation between circular points shows the speed of movement along each trajectory.

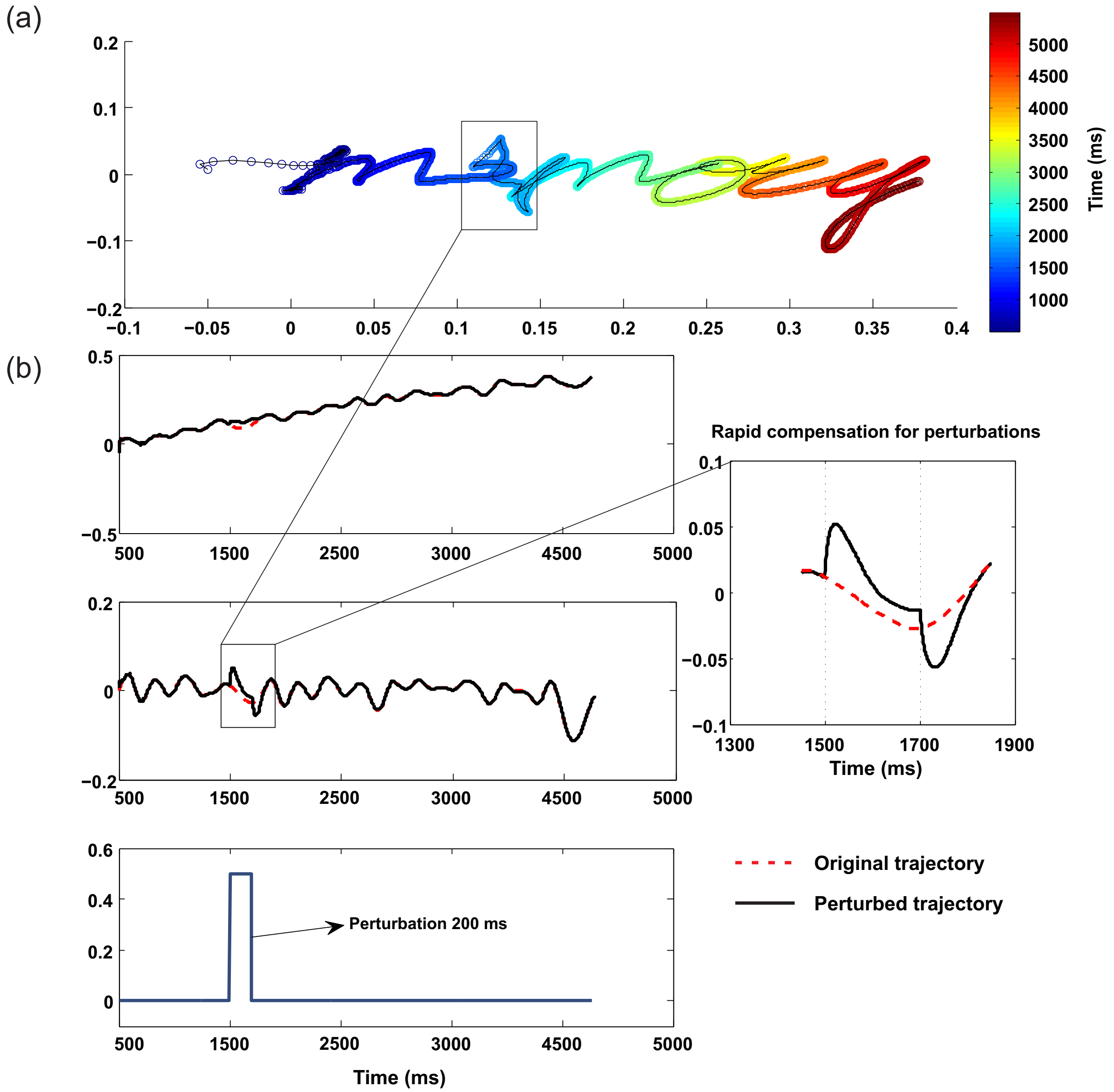


Figure 3.16: **Stable motor pattern generation in the presence of external perturbations**((a) Learned response of the reserovir network for generating the word 'memory' in the presence of a brief perturbation of high amplitude. The color coding shows the time of each event. The perturbation occurs at the time of the letter 'e', however the network can quickly compensate for the perturbation and return to the original trajectory. (b) The x and y coordinates (2-dimension time series) of the desired motor pattern (dotted red line) and the learned response in the presence of perturbation signal (solid black line). The zoomed in plot shows that the perturbation knocks the reservoir output of the actual trajectory, however it is able to quickly compensate for this and return to a stable path along the desired time series. The bottom plot shows the brief 200 ms perturbation signal given at 1500 ms time point. This was approximately at the time of the letter 'e'.

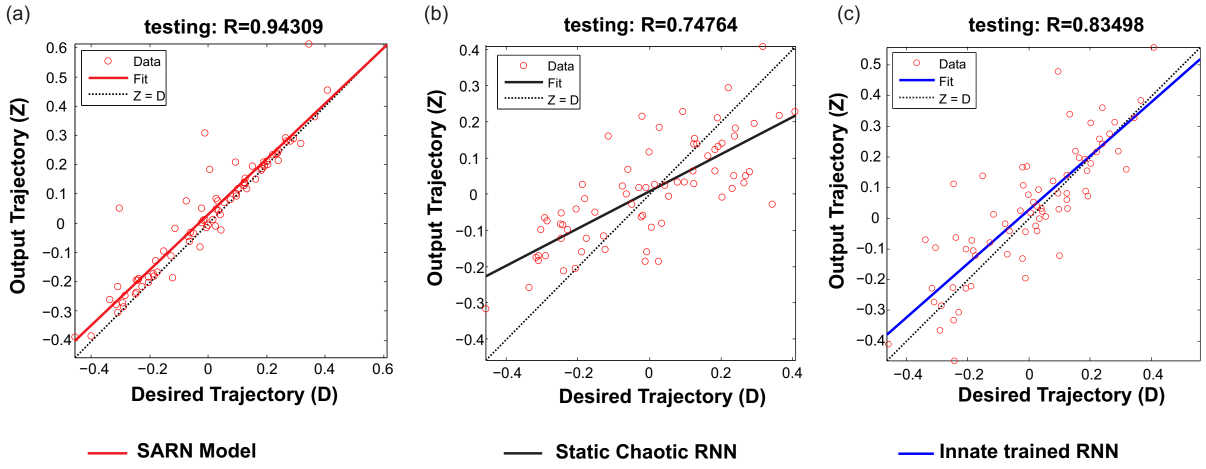


Figure 3.17: **Optimal performance with the SARN model under high noise condition**((a) Almost optimal linear relationship (regression co-efficient $R = 0.94$) between the learned output trajectory from SARN and the actual desired trajectory for the word 'memory', calculated over 50 trials with 5% noise in each reservoir neuron. (b) Considerable mismatch between the learned trajectory and the desired trajectory in case of the static chaotic RNN model (Sussillo and Abbott, 2009), (Jaeger and Haas, 2004). Significanlty low $R = 0.75$ (c) Learned trajectory with innate trained RNN model (Laje and Buonomano, 2013) considerably closer to the desired pattern ($R = 0.83$), however still worse as compared to SARN. Similar results were also obtained for the other motor pattern (learning to write 'apple').

In the presence of relatively high levels of noise (5% noise level), we carried out 50 trials to learn the trajectories for both the words using each of the networks. Regression analysis on the learned trajectories as compared to the actual handwritten patterns clearly demonstrate the superior performance of SARN as compared to both static chaotic RNNs as well as the recently introduced innate trained RNN (Fig. 3.17). SARN produced a near optimal linear fit with a regression coefficient of $R > 0.9$ accross all the 50 trials. In comparison the static network performs considerably poor with $R = 0.75$. Such behavior was expected based on our previous results where in SARN consistently outperforms static networks, proving that local adaptation and intrinsic plasticity mechanisms in combination with supervised synaptic plasticity is crucial for optimal temporal information processing. However the results indicate that supervised training of the reservoir neurons towards their innate trajectory as suggested by Laje and Buonomano (2013) does improve the performance in terms of stability of the learned trajectory and noise robustness (we found a regression coefficient of $R = 0.83$). However SARN with timescale adaptation and IP still outperforms. This suggests that a balance between homeostatic mechanisms and synaptic plasticity is an essential component of processing time varying stimuli and also generation of complex motor patterns. Although specialized supervised mechanisms to learn reservoir recurrent connections can be designed, their singular implementation still remains sub-optimal to a combination of homeostatic and synaptic plasticity, as present in SARN.

3.4 Discussion

3.4.1 Biological Relevance

The ability to precisely track and tell time is critical towards the learning of ordered motor behaviors as well as the underlying cognitive process, in all living creatures. However, the mechanism by which the brain performs robust temporal information processing is still not understood clearly. Although it is still debated whether dedicated or intrinsic mechanisms underlie the timing process in the brain, some experimental and theoretical studies have validated the concept of neural circuits being inherently capable of sensing time across time scales (Tetzlaff et al., 2012a), (Buhusi and Meck, 2005). Large recurrent neural networks like these reservoir systems could be considered as an abstraction of the mammalian cortex. Accordingly (Buonomano and Laje, 2010) suggested the concept of population clocks, where in time encoded in the time varying patterns of activity of neuronal populations emerge from the internal dynamics of the recurrent network. It is important to note that continuous input signals to these recurrent networks or the brain, in general can contain many different time-scales. In order to account for varying time-scales of input patterns to such networks, classically they have been setup in an hierarchical arrangement with different pre-determined timescales for each layer of hierarchy (Jaeger et al., 2007), (Yamashita and Tani, 2008). However monkey experiments (Bernacchia et al., 2011) have shown that individual neurons can have different timescales of reward memory correlated with the actual behavior. As such it is highly plausible that neurons in a single recurrent network can adjust or tune there individual time constants to account for a multi-timescale input in contrast to a hierarchical arrangement with different fixed timescales.

In this work, using a single information theoretic framework we have been successful in adapting the local neuron time constants via it's leak, while at the same time prevent runaway dynamical activity via the intrinsic plasticity mechanism. The combination of such homeostatic mechanisms with supervised synaptic plasticity in the reservoir network were also seen to lead to near critical dynamics, even when the network was initialized in the chaotic domain. Furthermore as observed in Figs. 3.11 (a), 3.12 (a), high local active information storage regions in the network correspond to significant events in time. According to the learning rule from equation 2.19, the individual neuron leak rates (time constants) have been adjusted according to the change of their AIS values with respect to a predefined threshold. In other words we were able to incorporate a self-adapting non-uniform neuron decay rate in the network that can account for varying timescales in the input stream as well as encode timing of events. As such in this work we not only present a mechanism to achieve a self-adaptive reservoir that can achieve a high degree of delayed temporal memory capacity, near critical dynamics and robustness to noise. From a biological perspective we show that time is not only encoded in the internal recurrent dynamics but also single neurons may adjust their time-constants in order to account for high relevance events in the input data.

3.4.2 Summary

In this chapter as continuation from chapter 2 we have presented and evaluated a self-adaptation mechanism for the reservoir network that successfully combines an intrinsic plasticity rule using a generic probability distribution (Weibull), with a reservoir neuron timeconstant (decay rate) adjustment rule based on input-driven local active information storage measure. The neuronal decay rates not only governs the degree of influence of local memory, but also collectively control the speed or timescale of the reservoir network dynamics. Due to feedback connections in such recurrent networks, chaotic or runaway activity had been previously observed in the works of (Sompolinsky et al., 1988) and (Sussillo and Abbott, 2009). The intrinsic plasticity mechanism ensures information maximization at each neurons output, while homeostatically regulating the network activity and preventing runaway chaotic dynamics. In general, our mechanism allows minimal parameter tuning, with two of the important network parameters decay-rates (τ_i), shape and scaling properties of neurons transfer function adjusted on the fly, in an unsupervised adaptive manner. In contrast, most static reservoirs pre-fix these parameter values or adapt them based on output error gradients that do not take into account difference in timescales of the input signal. Furthermore, by successfully combining the IP homeostatic rule, neuron timescale adaptation and the supervised synaptic plasticity in the recurrent and readout layers of the reservoir, we shed light on the importance of self-organized plasticity mechanisms in the brain that contribute towards its temporal information processing capabilities. The evaluated performance on the standard benchmark tasks and the complex multiple attractor learning tasks demonstrates that our adaptation mechanism clearly outperforms static reservoirs, while being able to modulate the reservoir dynamics in a input dependent manner. Moreover, we demonstrate the application of our network to generate clock like behaviors and the control of autonomous robotic agents through the maze navigation experiments, inherently requiring precise timing and delay memory. Finally using the complex motor pattern generation task, we demonstrated how dynamic attactors can be formed based on contextual inputs, that lead to specific motor patterns in a noise robust manner. It has been widely accepted that timing of events and memory guided behavior are intrinsically related. Specially for memory in the shorter time-scale of seconds to minutes (working memory), the system needs the ability to recognize important events in time. We achieve this in our network via the crucial combination of generic intrinsic plasticity and a novel neuron timescale-adaption that allows the neurons to speed up or slow down their dynamics based on the incoming input, while at the same time encode highly relevant events using the active information storage measure. Overall based on the methods from previous chapter and the in depth results obtained here, we motivate and demonstrate SARN as an adaptive input-driven RNN that forms a general model of temporal information processing in the brain, specifically in the timescale of few milliseconds to minutes.

CHAPTER 4

RESERVOIR-BASED ADAPTIVE FORWARD INTERNAL MODEL FOR COMPLEX MOTOR PREDICTION

> ”It is far better to foresee even without certainty than not to foresee at all”.
>
> —*Henri Poincare, The Foundations of Science.*

Motor prediction and planning is an integral outcome of robust temporal information processing in the brain. Since sensory information is substantially delayed, it has been proposed that the brain makes use of an internal forward model (Jordan and Rumelhart, 1992), (Wolpert et al., 1995), that can integrate both sensory and motor feedback signals to make precise predictions of current and upcoming body movements. Typically, forward model based timed motor responses occur on a timescale of milliseconds to seconds, while interacting with complex non-static environmental conditions (eg. motor prediction during walking on flat terrains differ significantly from predictions on irregular terrain). Therefore, such internal models not only require an intrinsic memory of recently issued motor commands, but also need the ability to adapt with changes in time varying sensory feedback signals. With this perspective, in this chapter, we demonstrate the ability of our self-adaptive RNN to work as internal forward models and generate complex locomotive behaviors. Specifically, taking inspiration from motor behaviors and internal models, observed in invertebrates (Webb, 2004), we present a neural mechanism to combine motor patterns generated by the central nervous system with our adaptive reservoir forward model (Manoonpong et al., 2014). This is implemented on a biologically inspired insect-like walking robot.

4.1 Introduction

Walking animals show diverse locomotor skills to deal with a wide range of terrains and environments. These involve intricate motor control mechanisms with internal prediction systems and learning (Huston and Jayaraman, 2011), allowing them to effectively cross gaps (Blaesing and Cruse, 2004), climb over obstacles (Watson et al., 2002), and even walk on uneven terrain (Pearson and Franklin, 1984), (Cruse, 1976). These capabilities are realized by a combination of biomechanics of their body and neural mechanisms. The main components of the neural mechanisms include central pattern generators (CPGs), internal forward models, and limb-reflex control systems. The CPGs generate basic rhythmic motor patterns for locomotion, while the reflex control employs direct sensory feedback (Pearson and Franklin, 1984). However, it is argued that biological systems need to be able to predict the sensory consequences of their actions to be capable of rapid, robust, and adaptive behavior. As a result, similar to the observations in vertebrate brains (Kawato, 1999), insects can also employ internal forward models as a mechanism to predicts their future (predictive feedbacks) state given the current state (sensory feedback) and the control signals (efference copies), in order to shape the motor patterns for adaptation (Webb, 2004).

In order to make such accurate predictions of future actions to satisfy changing environmental demands, the internal forward models (Fig. 4.1) needs memory of previous sensory-motor information. However, given that, such motor control happens on a very fast timescale, keeping track of temporal information is integral to such very short-term memory processes. Reservoir based RNNs (Maass et al., 2002), (Sussillo and Abbott, 2012) with their intrinsic ability to deal with temporal information and fading memory of sensory stimuli, thus provides the perfect platform to model such internal predictive mechanisms. Therefore we design SARN (Dasgupta et al., 2013a) (chapter 2) to act as the forward models that can work in conjunction with other neural mechanisms for motor control and generate complex adaptive locomotion in an artificial walking robotic system. Specifically, by exploiting the recurrent layer of our model it is possible to achieve complex motor transformations at different walking gaits, which cannot be achieved by currently existing simple forward models employed with walking robots (Manoonpong et al., 2013b), (Dearden and Demiris, 2005), (Schröder-Schetelig et al., 2010).

We present for the first time a distributed forward model architecture using six SARN-based forward models on a hexapod robot, each of which is for sensory prediction and state estimation of each robot leg. The outputs of the models are compared with foot contact sensory signals (feedback) and the differences between them are used for motor adaptation. This is integrated as part of the neural mechanism framework consisting of 1) central pattern generator-based control for generating basic rhythmic patterns and coordinated movements, 2) the reservoir forward models and 3) searching and elevation control for adapting the movement of an individual leg to deal with different environmental conditions.

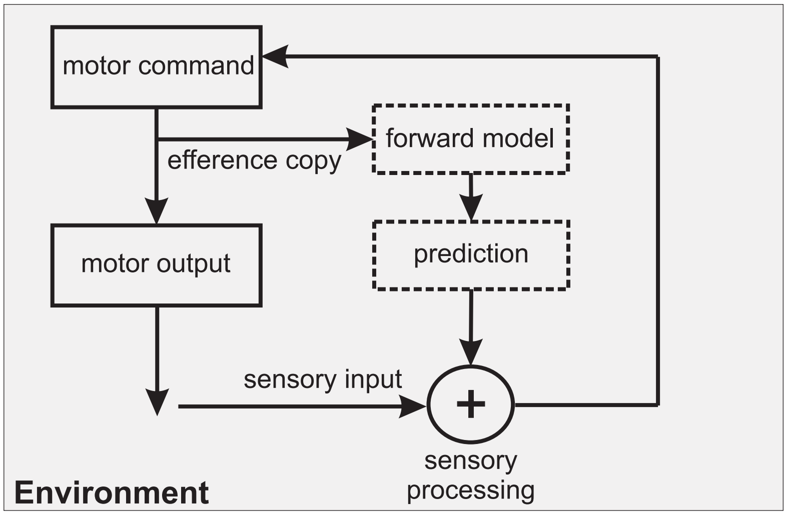


Figure 4.1: **Schematic representation of a sensory-motor system with a forward model** The solid arrows indicate the loop by which a motor command is translated into motor output, producing changes in the environment, which in turn causes changes in sensory input. This acts as feedback to the motor system to process further. The forward model is an internal loop that takes a copy of the motor command, and predicts the expected sensory input, which can be compared with the current sensory input to modulate behavior. A classical example is that moving our eyes causes the image on the retina to move, but we perceive a stable world because the image movement is predictable from the eye movement command (Webb, 2004).

4.2 Neural Mechanisms for Complex Locomotion

The neural mechanisms (Fig. 4.2a) are developed based on a modular structure. The mechanisms comprise i) central pattern generator (CPG)-based control, ii) reservoir-based adaptive forward models, and iii) searching and elevation control. The CPG-based control and the searching and elevation control have been discussed in detail in Manoonpong et al. (2013b), thus here we will only provide a brief overview of these mechanisms, while the reservoir-based adaptive forward models, which is the main topic of this chapter, will be presented in detail in the following section.

The CPG-based control primarily generates a variety of rhythmic patterns and coordinates all leg joints of a hexapod robot AMOSII (Fig. 4.2 (b)), thereby leading to a multitude of different behavioral patterns and insect-like leg movements. The patterns include omnidirectional walking and insect-like gaits (Manoonpong et al., 2013b). All these patterns can be set manually or autonomously driven by exteroceptive sensors, like a camera (Zenker et al., 2013), a laser scanner (Kesper et al., 2013), or range sensors. While the CPG-based control provides versatile autonomous behaviors, the searching and elevation control using the accumulated error signals provided by the reservoir-based adaptive forward models adapts the movement of an individual leg of the robot to deal with different environmental conditions.

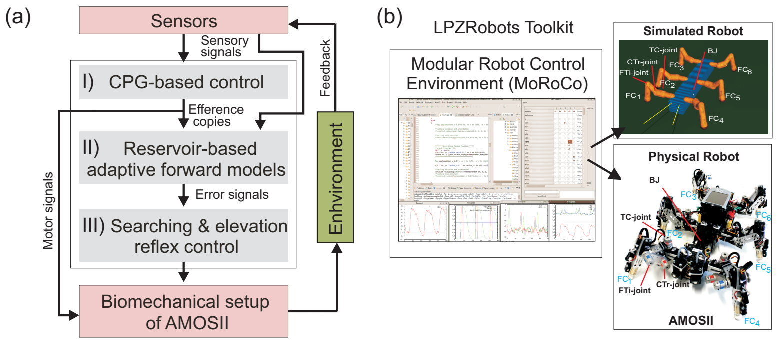


Figure 4.2: (a) The diagram of an artificial bio-inspired walking system consisting of the biomechanical setup of the hexapod robot AMOSII (i.e., six 3-jointed legs, a segmented body structure with one active backbone joint (BJ), actuators, and passive compliant components (Manoonpong et al., 2013b)), sensors (i.e., proprioceptive and exteroceptive sensors), and neural mechanisms (i,ii,iii). (b) Modular Robot Control Environment embedded in the LPZRobots toolkit. It is used for developing a controller, testing it on the simulated hexapod robot, and transferring it to the physical one. FC_1, FC_2, FC_3, FC_4, FC_5, and FC_6 are foot contact sensors installed in the robot legs. Each leg has three joints: the thoraco-coxal (TC-) joint enables forward and backward movements, the coxa-trochanteral (CTr-) joint enables elevation and depression of the leg, and the femur-tibia (FTi-) joint enables extension and flexion of the tibia. The morphology of these multi-jointed legs based on a cockroach leg (Zill et al., 2004). More details on BJ control for climbing can be found in (Goldschmidt et al., 2014).

This CPG-based control (see appendix A.3) itself is designed as a modular neural network that consists mainly of four elements:

1. CPG mechanism with neuromodulation for generating different rhythmic signals. Inspired by biological findings, here the CPG circuit is designed as a two neuron fully connected recurrent network (Pasemann et al., 2003), such that using external neuromodulatory inputs different walking gates can be achieved.
2. CPG post-processing units (PCPG) for shaping CPG output signals.
3. Phase switching network (PSN) and velocity regulating networks (VRNs) for walking directional control.
4. Motor neurons with delay lines (delay λ) for transmitting motor commands to all leg joints of AMOSII. These delay lines are utilized to realize the inter-limb coordination, in which they introduce phase differences between the transmitted signals to all leg joints. As a result, the desired gait is achieved.

The searching and elevation reflex control consist of single recurrent neurons that received the difference (instantaneous error) between the predicted forward model signal and the actual sensory feedback. Due to the recurrency, this is accumulated over time. The accumulated error can then be used to either extend specific leg joints in order to get better foothold (searching reflex) during stance phase or elevated further to overcome obstacles during the swing phase (see Fig. 4.7 in section 4.4.1). All neurons in the CPG-based control and the searching and elevation control are modeled as discrete-time non-spiking neurons with tan-hyperbolic or piecewise linear activation functions (see (Manoonpong et al., 2013b) for details). They are updated with a frequency of ≈ 27 Hz.

4.3 Materials & Method

4.3.1 Reservoir-based Distributed Adaptive Forward Models

Six identical adaptive forward models ($RF_{1,2,3,\ldots,6}$) are used here, one for each leg (Fig. 4.3(a)). They serve for sensory prediction as well as state estimation. Specifically, each forward model transforms an efference copy of the actual motor signal for one of leg joints (i.e., here the CTr-motor signal[1]), into an expected or predicted sensory signal. This can be then compared with the actual incoming sensory feedback signals (i.e., here the foot contact signal - Fig. 4.3 (b), of each leg) and, based on the error, trigger the appropriate reflex (searching or elevation) and modulate the behavior of the robot.

Each forward model is based on the self-adaptive reservoir network (SARN - as introduced in chapters 2 and 3) (Dasgupta et al., 2013a) type. As exhibited in the previous chapter, due to the dynamic reservoir and homeostatic adaptations, the network exhibits a wide repertoire of nonlinear activity and long fading memory. This can be exploited for the motor signal transformation and motor prediction needed in the current context.

Network Setup

Using a typical construction, each reservoir forward model consisted of three layers: input, hidden (or internal), and readout layers (Fig. 4.3 (b)). The internal layer is constructed as a random RNN with N internal neurons and a fixed randomly initialized synaptic connectivity (in this setup we only modify the reservoir-to-readout neuron weights).

Here we use a discrete time version of the original SARN formulation (Eq. 2.14), such that using $\Delta t = 1$, the discrete time state dynamics of each reservoir neuron is given by:

$$x_i(t+1) = \left(1 - \frac{1}{\tau_i}\right) x_i(t) + \frac{1}{\tau_i}\left(g \sum_{j=1}^{N} W_{i,j}^{rec} r_j(t) + W_{i,1}^{in} u(t) + B_i\right), \tag{4.1}$$

$$\mathbf{z}(t) = \mathbf{W}^{out}\mathbf{x}(t), \tag{4.2}$$

---

[1] We use the CTr-motor signal instead of the TC- and FTi-motor signals since this shows clear swing (off the ground) and stance (on the ground) phases which can be simply matched to the foot contact signal.

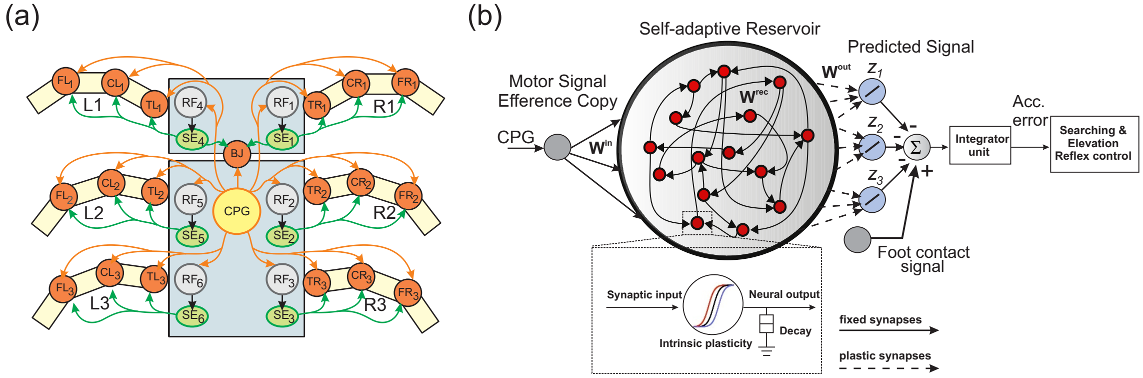


Figure 4.3: (a) Neural mechanisms implemented on the bio-inspired hexapod robot AMOSII. The yellow circle (CPG) represents the neural locomotion control mechanism (see appendix. A.3). The gray circles ($RF_{1,2,3,\ldots,6}$) represent the reservoir-based adaptive forward models. The green circles ($SE_{1,2,3,\ldots,6}$) represent searching and elevation control modules. The orange circles represent leg joints where TR_i, CR_i, FR_i are TC-, CTr- and FTi-joints of the right front leg ($i = 1$), right middle leg ($i = 2$), right hind leg ($i = 3$) and TL_i, CL_i, FL_i are left front leg ($i = 1$), left middle leg ($i = 2$), left hind leg ($i = 3$), respectively. BJ is a backbone joint. The orange arrow lines indicate the motor signals which are converted to joint angles for controlling motor positions. The black arrow lines indicate error signals. The green arrow lines indicate signals for adapting joint movements to deal with different circumstances. b) An example of the reservoir-based adaptive forward model. The dashed frame shows a zoomed in view of a single reservoir neuron. In this setup, the input to each of the reservoir network comes from the CTr- joint of the respective leg. The reservoir learns to produce the expected foot contact signal for three different walking gaits (z_1, z_2, z_3). The signals of the output neurons are combined and compared to the actual foot contact sensory signal. The error from the comparison is transmitted to an integrator unit. The unit accumulates the error over time. The accumulated error is finally used to adapt joint movements through searching and elevation control.

where all the variables reflect the same quantities as introduced previously in chapter 2, section 2.2.1, however no explicit feedback from the readout neurons is present.

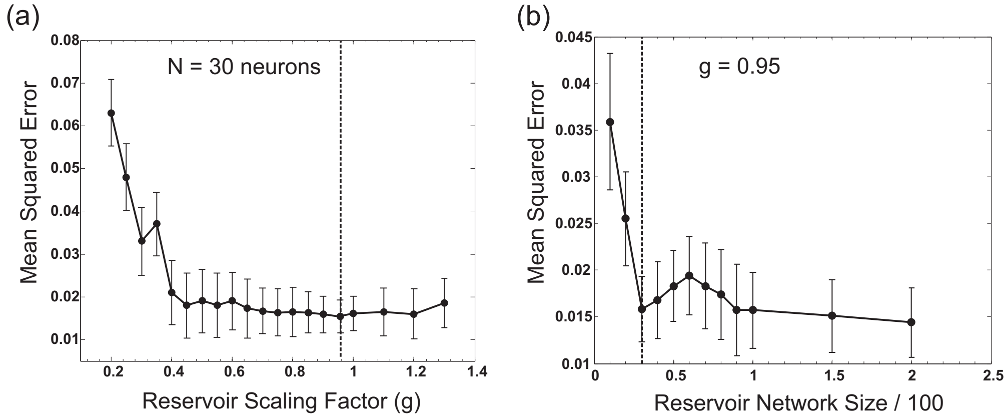


Figure 4.4: (a) Plot of the change in the mean squared error for the forward model task for one of the front legs (R_1) of the walking robot with respect to the scaling of the reservoir weight matrix with different g. As observed, very small values in g have a negative impact on performance compared with values closer to one being better. Interestingly, the performance did not change significantly for $g > 1.0$ (chaotic domain). This is mainly due to homeostasis introduced by intrinsic plasticity in the network. The optimal value of $g = 0.95$ selected for our experiments is indicated with a dashed line. (b) Plot of the change in mean squared error with respect to different reservoir sizes (N). g was fixed at the optimal value. Although increasing the reservoir size in general tends to increase performance, a smaller size of $N = 30$ gave the same level of performance as $N = 100$. According for computational efficiency, we set our reservoir size to 30 neurons. Results were obtained from 10 trials with different parameter initializations on the forward model task for a single leg and a fixed walking gait.

The input to the reservoir $u(t)$, consisted of a single CTr-motor signal. This acts as an efference copy of the post-processed CPG output. The readout layer consisted of three neurons, with their activity being represented by the three-dimensional vector $\mathbf{z}(t)$. Although typically $M < N$ readout neurons can be connected to the reservoir, here we restricted it to three neurons, as each readout here, learns the predictive signal for one of the following different walking gaits: wave (z_1), tetrapod (z_2), and caterpillar (z_3) gaits). The wave, tetrapod, and caterpillar gaits are used for climbing over an obstacle, walking on uneven terrain, and crossing a large gap[2], respectively. Subsequent to the supervised training of the reservoir-to-readout connections $\mathbf{W}^{out}$, each readout neuron basically learns to generate the expected foot contact signal associated with each gait. The decay rate for each reservoir neuron is given by $\frac{1}{\tau_i}$, where τ_i is the individual membrane timeconstant. The input-to-reservoir connections weights $\mathbf{W}^{in}$ and internal recurrent weights $\mathbf{W}^{rec}$ were drawn randomly from the uniform distribution $[-0.1, 0.1]$ and a Gaussian

---

[2] These three gaits were empirically selected among 19 others. Previous studies show that wave and tetrapod gaits are the most effective for climbing and walking on uneven terrains, respectively. While in this study we observed that the caterpillar gait was the best one for crossing a gap. However, without any loss of performance, more gaits can be applied easily by adding further output neurons.

distribution of zero mean and variance $\frac{1}{pN}$, respectively. In order to select the appropriate reservoir size, empirical evaluations were carried out, such that a moderate network size of $N = 30$ was selected, for which minimum prediction error was obtained at the output layer. The recurrent weights were subsequently scaled by the factor of $g = 0.95$ (see Fig. 4.4). Each reservoir neuron were updated with a frequency of ≈ 27 Hz using a tanh nonlinear activation function, $r_i(x_i) = tanh(a_i x_i + b_i)$. As described in Sections 2.2.2 and 2.2.3, intrinsic plasticity and neuron timescale adaptation were carried out in order to learn the transfer function and reservoir timeconstant parameters.

Here we used a slightly modified version of Eq. 2.32 based on the original recursive least squares (RLS) algorithm (Jaeger and Haas, 2004),(Simon, 2002) in order to learn the output weights $\mathbf{W}^{out}$ at each time step, while the training input $u(t)$ is being fed into the reservoir. $\mathbf{W}^{out}$ are calculated such that the overall error is minimized; thereby the network transforms the CTr-motor signal to the expected foot contact signal correctly. We implement the RLS algorithm using a fixed forgetting factor ($\lambda_{RLS} < 1$) as follows:

$$e(t) \leftarrow d(t) - \sum_{j=1}^{3} z_j(t), \tag{4.3}$$

$$\mathbf{K}(t) \leftarrow \frac{\rho(t-1)\mathbf{r}(t)}{\lambda_{RLS} + \mathbf{r}^T(t)\rho(t-1)\mathbf{r}(t)}, \tag{4.4}$$

$$\mathbf{P}(t) \leftarrow \frac{1}{\lambda_{RLS}}\Big[\mathbf{P}(t-1) - \mathbf{K}(t)\mathbf{r}^T(t)\rho(t-1)\Big], \tag{4.5}$$

$$\mathbf{W}_{out}(t) \leftarrow \mathbf{W}_{out}(t-1) + \mathbf{K}(t)e(t). \tag{4.6}$$

Parameter	Symbol	Value
Reservoir size	N	30
Reservoir neuron noise	B_i	$\in N(0, 0.001)$
Neuron timeconstant (initialization)	τ_i	1.0
RLS learning constant	δ_c	10^{-4}
Non-linearity shape initialization	a_i	1.0
Non-linearity scale initialization	b_i	0.0
RLS learning rate	λ_{RLS}	0.99
Reservoir connection probability	p	0.5
Scaling parameter	g	0.95
Input weights	$\mathbf{W}^{in}$	$\in$ U$[-0.15, 0.15]$
Reservoir weights	$\mathbf{W}^{rec}$	$\in N(0, 1/pN)$

Table 4.1: The list of Reservoir network parameter settings

Here $e(t)$ is the online error calculated from the difference between the desired output, $d(t)$ (i.e., here expected foot contact signal) and the summation of all generated reservoir readouts (predicted output). $\mathbf{K}(t)$ is the RLS gain vector and $\mathbf{P}(t)$ the inverse correlation matrix of reservoir neuron firing rate, updated at each time step. The reservoir to readput weights $\mathbf{W}^{out}$ is initially set to zero. Exponential forgetting factor (λ_{RLS}) is set to a value less than one (here, we use 0.99). The inverse correlation matrix $\mathbf{P}$ is initialized as $\mathbf{P}(0) = \mathbf{I}/\delta_c$, where $\mathbf{I}$ is the unit matrix and δ_c is a small constant (here, $\delta_c = 10^{-4}$). Details of all the fixed parameters and initial settings for the reservoir based forward model networks are summarized in Table 4.1.

4.4 Results

4.4.1 Learning the Reservoir Forward Model (motor prediction)

In order to train the six forward models ($RF_1 to RF_6$) in an online manner, one for each leg, we let the simulated robot AMOSII walk under normal condition (i.e., walking on a flat terrain with the three different gaits). Initially, we let the robot walk with a certain gait, and then every 2500 time steps, the gait pattern was sequentially altered (this occurs by changing the modulatory input to the CPG). As a result, the robot sequentially transitions from wave gait, to tetrapod gait, to caterpillar gait repeatedly. Using this procedure, we let the robot walk for three complete cycles (22500 time steps) and collected the corresponding CTr-motor signal and foot contact sensor readings for all legs. Intrinsic plasticity and neuron time constant adaptations were then carried out using 20 epochs of 1000 time steps overlapping time windows. After this pre-training phase, all the reservoir neuron non-linearity parameters and individual time constants (τ_i) were fixed.

Subsequent to the pre-training phase, normal training of the reservoir-to-readout weights $\mathbf{W}^{out}$ was carried out using the online RLS learning algorithm with the same process of making the robot walk on a flat, regular terrain and sequential switching between the three gait patterns every 2500 time steps. As such, at any given point in time only one of the readout neurons (specific to the walking gait) are active. In this manner, synaptic weights projecting from reservoir to the first readout neuron (y_1) corresponding to the foot contact signal prediction for the wave gait, and synaptic weights projecting to the second (y_2) and third (y_3) readout neurons corresponding to the foot contact signal prediction of the tetrapod and caterpillar gaits, are learned, respectively. In this experimental setup, as observed from Fig. 4.5 (a), (b) and (c) the readout weights corresponding to each gait converges very quickly, in less than the trial period of 2500 time steps. As a result, every time the CTr-motor signal changes due to walking gait transformations, the RF associated with each leg learns to predict the expected foot contact signal robustly. The training process was carried out only once under normal walking conditions. This was subsequently used as the baseline in order to compare with the actual foot contact signals (sensory feedback) while walking under the situations of crossing a gap, climbing, and negotiating uneven terrains.

Fig. 4.6 shows an example of the forward model prediction (training) during the three different gaits for the right front leg of AMOSII (R_1). Visual inspection clearly demonstrates that according to the corresponding efference copy of CTr-motor signal at a particular gait, the expected foot contact (FC) signal is precisely predicted at each time point. Similarly, the foot contact signals for the other legs are also predicted online, given the current context of CTr-signal (not shown). Note that the FC signals of the other legs normally show slightly different periodic patterns. Furthermore, there exists considerable lag between the expected stance phase according to the motor signal and that observed from the FC signal (difference between dotted green lines in Fig. 4.6). Due to the internal memory of the incoming motor signal in the reservoir, we see that the output neurons can adapt to these time lags efficiently, even when the frequency of the signal increases with a change in walking gaits. This was not possible in the previous state of the art single recurrent neuron forward models (Manoonpong et al., 2013b). As such, a simple square wave matching the timing of the motor signal efference copy was used, providing a limited range of behavior, as well as being biologically unrealistic. However, here our reservoir model can robustly learn the exact shape and timing of the FC signals.

During testing of the learned behavior, while AMOSII walks under different environmental conditions and a specific gait, the output of each trained forward model (i.e., the predicted FC signal, Fig. 4.7 (a)) is used to compare it to the actual incoming FC signal of the leg (Fig. 4.7 (b)). The difference (instantaneous error signal Δ) between them determines the walking state where a positive value ($+\Delta$) indicates losing ground contact during the stance phase and a negative value ($-\Delta$) indicates stepping on or hitting obstacles during the swing phase.

$$\Delta_i(t) = RF_i(t) - FC_i(t). \tag{4.7}$$

where $i \in \{1, 2, ..., 6\}$ represents each leg of the robot.

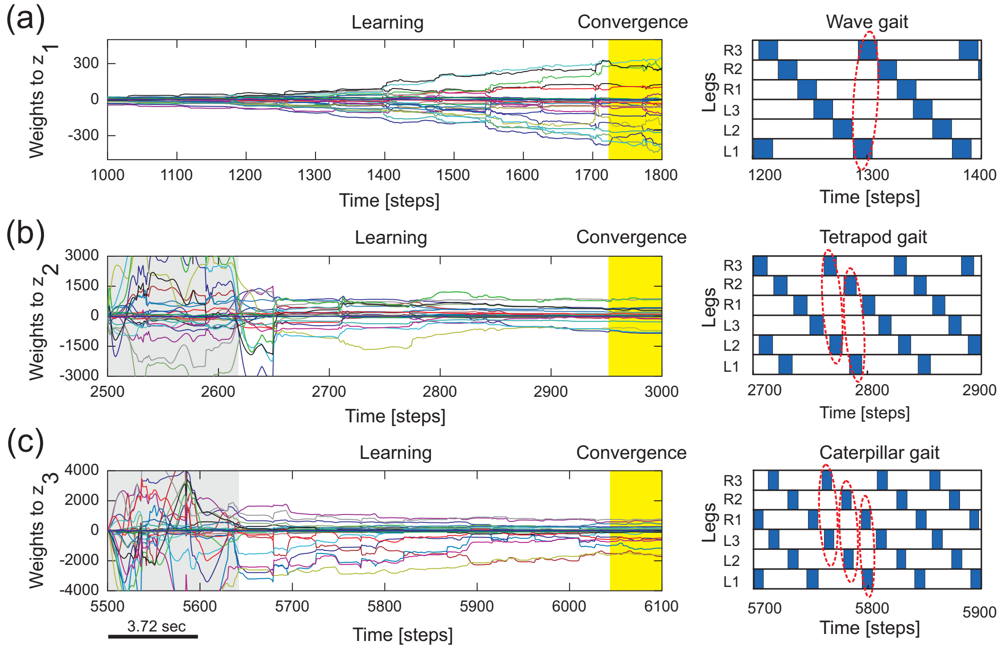


Figure 4.5: **Reservoir-to-readout weight adaptation during online learning.** (a) Changes of 30 weights projecting to the first readout neuron (z_1) of the forward model of the right front leg (R_1) while walking with a wave gait. During this period, weights projecting to the second (z_2) and third (z_3) output neurons remain unchanged (i.e., they are zero). (b) Changes of the weights to z_2 while walking with a tetrapod gait. During this period, the weights to z_3 still remain unchanged and the weights to z_1 converge to around zero. (c) Changes of the weights to z_3 while walking with a caterpillar gait. During this period, the weights to z_1 and z_2 converge to around zero. At the end of each gait, all weights are stored such that they will be used for locomotion in different environments. The grey areas represent transition phases from one gait to another gait and the yellow areas represent convergence. The gait diagrams are shown on the right. They are observed from the motor signals of the CTr-joints (Fig. 4.6). White areas indicate ground contact or stance phase and grey areas refer to no ground contact during swing phase. As frequency increases, some legs step in pairs (dashed enclosures). Here convergence implies no siginificant change in the vector norm of the readout weights.

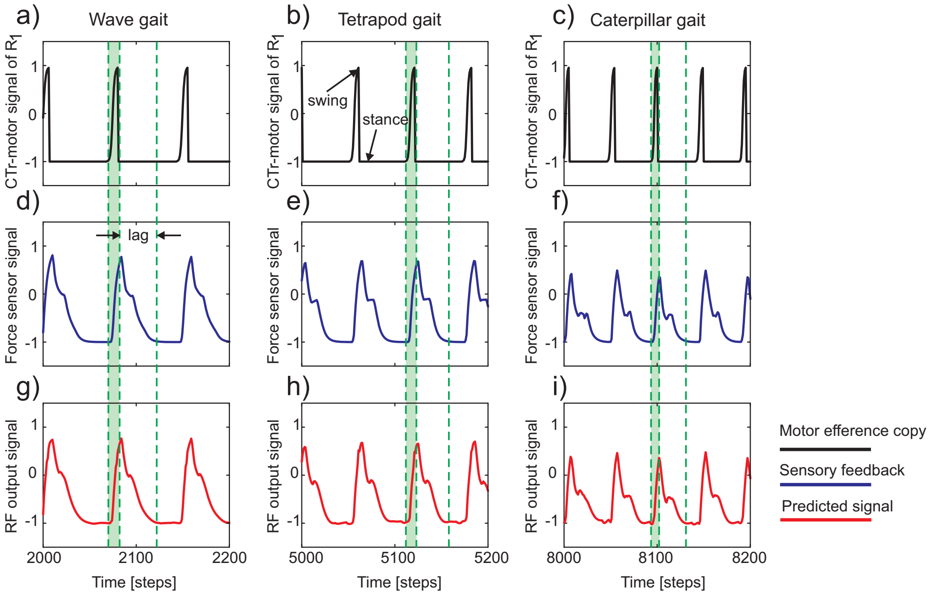


Figure 4.6: (a-c) The CTr-motor signal of the right front leg (R_1) for wave, tetrapod, and caterpillar gaits, respectively. This motor signal is basically the input of the forward model. (d-f) The foot contact signal (force sensor signal under normal walking conditions) used as the target signal of the reservoir network. (g-i) The predicted foot contact signal or the final output of the forward model (RF output signal). The green shaded region indicates the time between swing and stance phase for the CTr motor signal at the three walking gaits. As observed the actual foot contact signal is considerably lagged in time compared to the motor signal. Effectively, this lag decreases with an increase in the gait frequency. The single RF adaptively accounts for these different delay times in order to accurately predict the expected foot contact signal.

Thus, we use the positive value for searching control (Fig. 4.7 (d) above). This is then accumulated through a single recurrent neuron S with a linear transfer function and is always reset to 0.0 at the beginning of swing phase. Similarly, the negative value is used for elevation control (Fig. 4.7 (d) below). The value is also accumulated through a recurrent neuron E with a linear transfer function. These accumulated errors thus allow the robot leg to be either elevated (on hitting an obstacle) or searching for a foothold during the swing and stance phase respectively (see Manoonpong et al. (2013b) for more details of the searching and elevation control). As depicted in Fig. 4.7 (a) and (b), while walking on a rough terrain (in this case with tetrapod walking gait), the currently recorded sensory feedback or foot contact sensor reading differs considerably from the reservoir predicted signal. As a result, there is a high accumulation of error between each swing or stance phase (Fig. 4.7 (c)). This causes the corresponding reflex control mechanism to kick in and the robot successfully navigates out of the rough terrain (after ≈ 4000 time steps). Once the robot moves into the flat terrain, the reservoir predicted foot contact signal matches almost perfectly with the actual sensory feedback. As a result, the accumulated error becomes zero and normal walking without any additional reflex mechanism can continue. In the specific case of gap crossing, we use the accumulated error to control tilting of the backbone joint (BJ) and shifting of the TC- and FTi-joints such that the legs are extended forward continuously till the robot can find a foothold (see the experiments and results section below). For climbing and walking on uneven terrain, it causes shifting of the CTr- and FTi-joints such that the respective leg searches for a foothold. In addition to this leg joint control, reactive backbone joint control using the additional ultrasonic sensory in front of the robot can also be used to learn to lean up the BJ for climbing over obstacles (this has been previously successfully applied in Goldschmidt et al. (2014)).

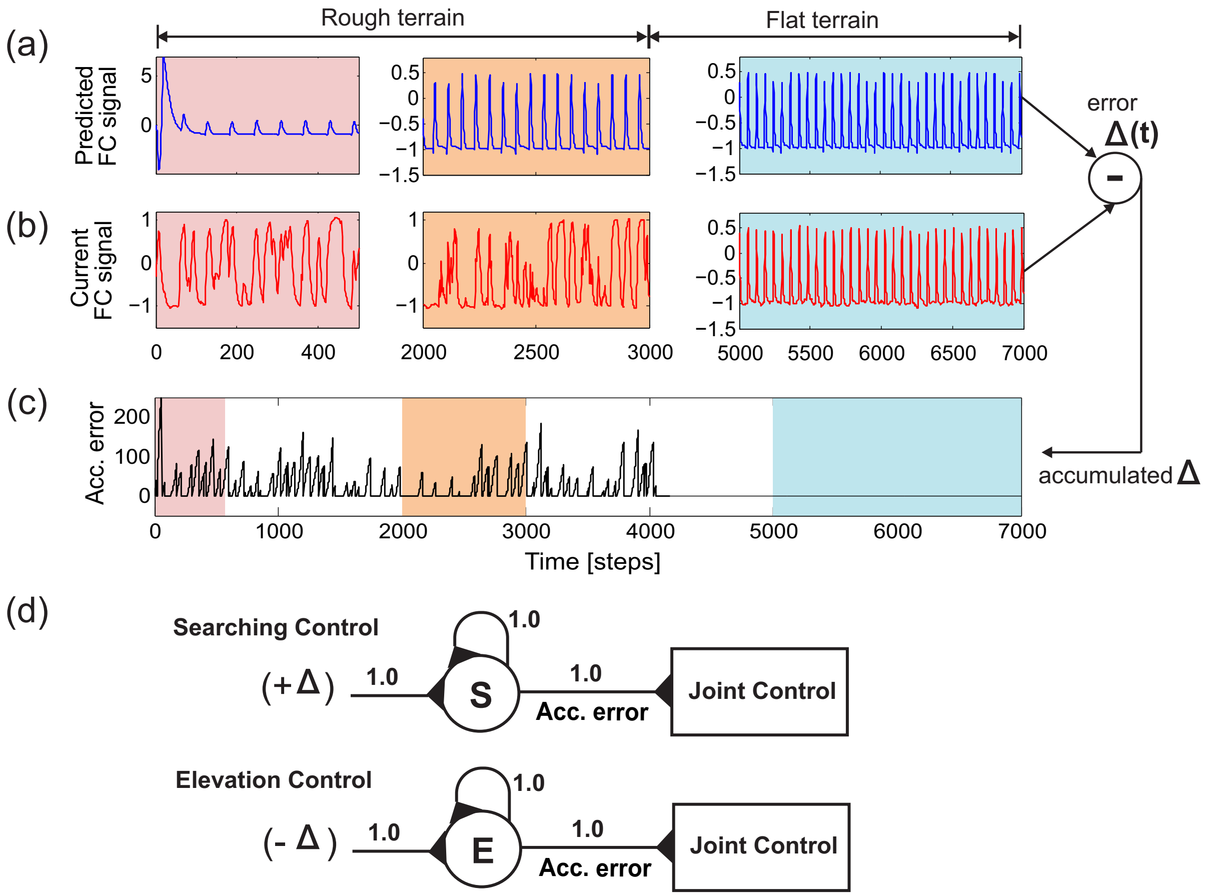


Figure 4.7: **Successfully navigating rough terrain with reservoir forward model** (a) The reservoir forward model predicted, expected foot contact signal. After a small initial transient the reservoir output quickly converges to the expect signal for normal walking condition. (b) The actual sensory feedback (foot contact signal) while walking on the rough surface (c) Accumulated error calculated from the instantenous error ($\Delta(t)$). (d) The searching and elevation reflex control. After 4000 time steps, the robot successfully overcomes the rough terrain and continuous walking on a flat surface. As a result, there is zero accumulated error since the predicted foot contact signal almost exactly matches the actual signal.

4.4.2 Simulation Results

In order to assess the ability of the reservoir-based forward models to generate memory[3] guided behaviors in a neural closed-loop control system (see Fig. 4.2), we conducted simulation experiments under different situations including crossing a gap, climbing over high obstacles, and walking on uneven terrain (similar to the behaviors observed in real insects). In all cases, we used the same learned forward model under normal walking conditions for a flat terrain (Section 4).

We now take the example of the gap crossing experiment in order to look in detail at the learning outcome of the forward models. For gap crossing, we let AMOSII walk with a caterpillar gait (see Fig. 4.5 (c), right), such that each left and right pair of legs moves simultaneously. As shown in Fig. 4.8(1), at the beginning AMOSII walked forward straight towards the gap. In this period, as it walks on the flat surface of the platform, it performed regular movements similar to the training period under normal walking conditions (first platform) . Afterward, it encountered a 15 cm gap (≈ 44% of body length - the maximum cross-able distance). In this situation, during the subsequent stance phase its front legs loose ground contact (Figs. 4.8(d) and (e)). As a result, the foot contact sensors from the front legs do not record any value. However the reservoir forward model still predicts the expected foot contact signal causing a positive instantaneous error (Eq. 4.7). This leading to a gradual ramping of the accumulated error signal between each stance phase and swing phase, for the front legs (Fig. 4.8 (a)).

In order to activate the BJ and adapt the leg movements due to the error signals, we used the maximum accumulated error value of the previous step (Fig. 4.8, (a) red line) and control the BJ and leg movements in the subsequent step. In this manner, the BJ started to lean upwards incrementally at around $1020 - 1170$ time steps (Fig. 4.8(2)). Simultaneously, the TC- and FTi-joint movements of the left and right front legs were also adapted accordingly. Due to a predefined time-out period for tilting upwards, at around 1170 time steps (Fig. 4.8(3)), the backbone joint automatically moved downwards. Consequently, the front legs touched the ground of the second platform at the middle of the stance phase; thereby, causing the accumulated error signals to decrease. Due to another time-out period for tilting downwards at around 1200 time steps (Fig. 4.8(4)), the BJ automatically moved to the normal position ($-2\,\text{deg}$). Since now the situation is similar to walking on flat terrain, the RF predicted foot contact signal matches the one recorded by the foot sensors, with accumulated error dropping to zero. Thereafter, the TC- and FTi-joints perform regular movements. At around 1300 time steps (Fig. 4.8(5)), the left and right hind legs loose the ground contact, leading to body tilting. As a result, the movements of the TC- and FTi-joints were slightly adapted allowing AMOSII to successfully cross the gap and continue walking on the second platform (Fig. 4.8(6)).

Fig. 4.9 shows that the reservoir forward model in combination with the neural locomotion control mechanisms, not only successfully generates gap crossing behavior of AMOSII (as shown above), but also allows it to climb over single and multiple obstacles (eg. up a fleet of stairs), as well as enables the robot to walk on uneven terrain. In all these cases, similar to we directly used the accumulated errors for movement adaptation via the searching and elevation control

---

[3] Forward models for motor prediction need an internal fading memory of the motor aparatus, in order to adjust for time delays between motor output signal and the actual sensory feedback (Kawato, 1999).

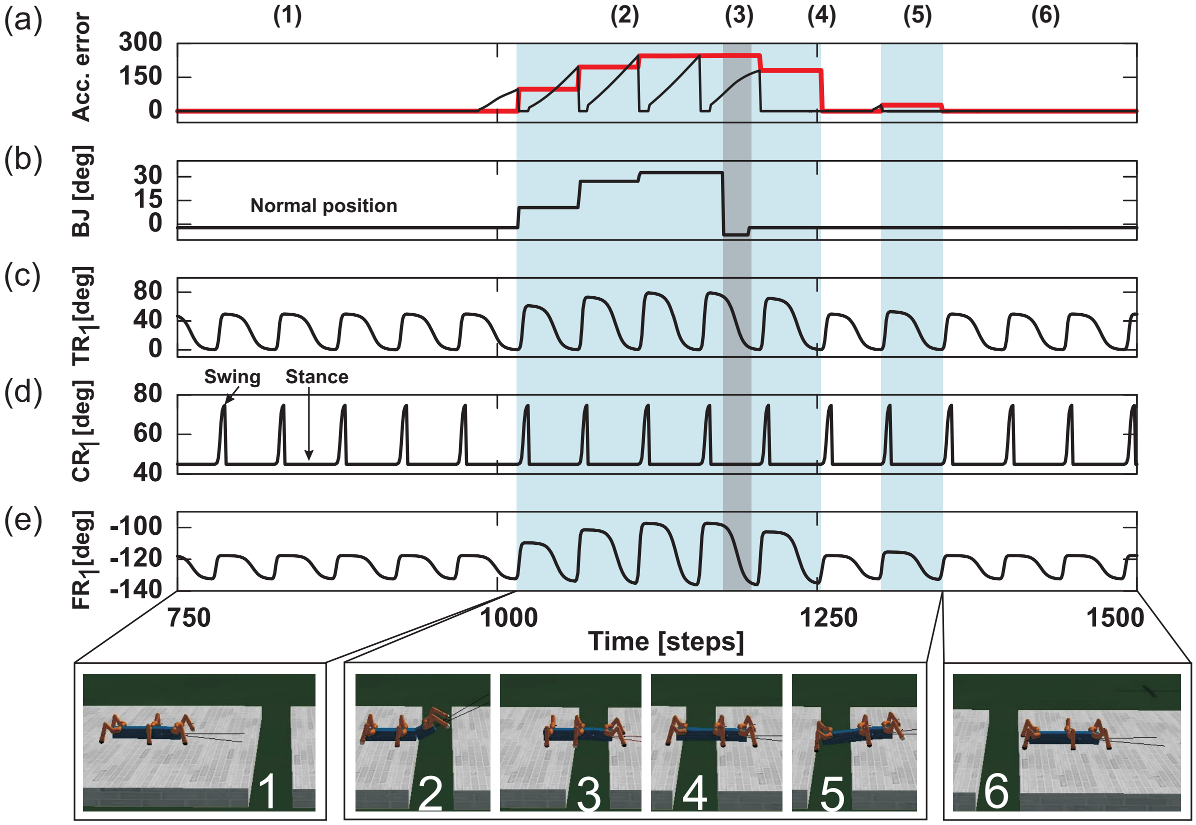


Figure 4.8: **Real-time data of walking and crossing a large gap using the forward model prediction.** (a) The accumulated error (black line) and the maximum accumulated error value at the end of each stance phase (red line) of the right front leg (R_1). The accumulated error is reset to zero every swing phase. (b) The backbone joint (BJ) angle during walking and gap crossing. The BJ stayed at a normal position (-2 deg) during normal walking. It leant upwards and then bent downwards during gap crossing. (c-e) The TC-, CTr-, and FTi-joint angles of R_1 during walking and gap crossing. The joint adaptation was controlled by the maximum accumulated error value of the previous step (red line). Below pictures show snap shots of the locomotion of AMOSII during the experiment. Note that one time step is ≈ 0.037 s.

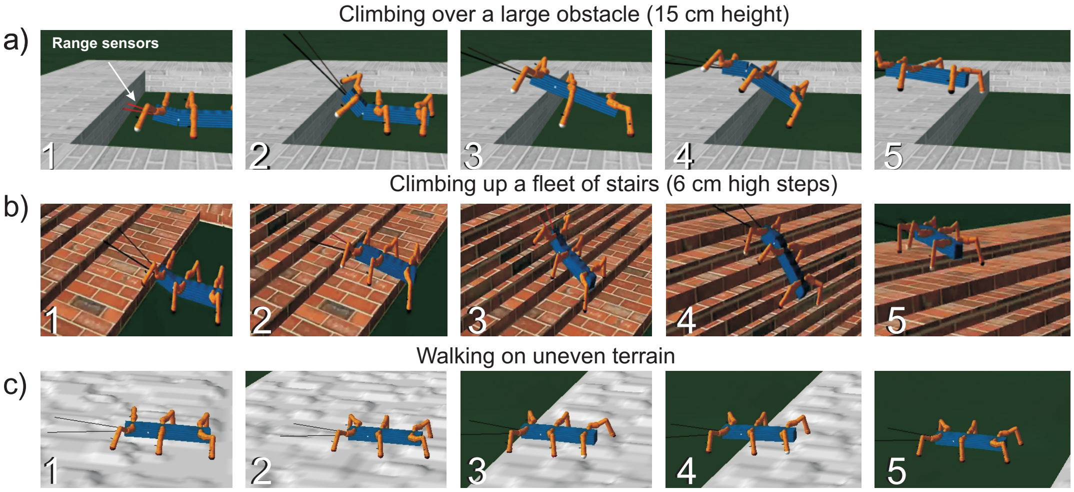

Climbing over a large obstacle (15 cm height)
a)
Range sensors
1
2
3
4
5
Climbing up a fleet of stairs (6 cm high steps)
b)
1
2
3
4
5
Walking on uneven terrain
c)
1
2
3
4
5

Figure 4.9: **Snap shots during climbing over a high obstacle, climbing up a fleet of stairs, and walking on uneven terrain.** (a) AMOSII walked with the wave gait and approached a 15 cm high obstacle (1). It detected the obstacle using its range sensors installed at its front part. The low-pass filtered range sensory signals control the BJ to tilt upwards (2) and then back to its normal position (3). Due to the missing foot contact of the front legs, the BJ moved downwards to ensure stability (4). During climbing, middle and hind legs lowered downwards due to the occurrence of the accumulated errors, showing leg extension, to support the body. Finally, it successfully surmounted the high obstacle (5) (see video at http://manoonpong.com/ComplexLocomotion/S2.wmv). (b) AMOSII climbed up a fleet of stairs (1-5) using the wave gait as well as the reactive BJ control. The climbing behavior is also similar to the one described in the case (a) (see video at http://manoonpong.com/ComplexLocomotion/S3.wmv). (c) AMOSII walked with the tetrapod gait. During traversing from the uneven terrain (1-4) to the even terrain (5), it adapted its legs individually to deal with a change of terrain. That is, it depressed its leg and extended its tibia to search for a foothold when losing a ground contact during the stance phase. Losing ground contact information is detected by a significant change of the accumulated errors (see video at http://manoonpong.com/ComplexLocomotion/S4.wmv).

mechanisms. For climbing, the reactive backbone joint control was also applied to the system (see Goldschmidt et al. (2014) for more details) and a slow wave gait walking pattern (see Fig. 4.5 (a), right).

Experimentally the wave gait was found to be the most effective for climbing, which allows AMOSII to overcome the highest climbable obstacle (i.e., 15 cm height which equals $\approx 86\%$ of its leg length) and to surmount a fleet of stairs. For walking on uneven terrain, a tetrapod gait (see Fig. 4.5 (b), right) was used without the backbone joint control. This is the most effective gait for walking on uneven terrain (see also (Manoonpong et al., 2013b)). Recall that in all experiments the forward models basically generate the expected foot contact signals (i.e., sensory prediction), which are compared to the actual incoming ones. Errors between the expected and actual signals during locomotion serve as state estimation and are used to adapt the joint movements accordingly. It is important to note that, the best gait for each specific scenario was experimentally determined and fixed. However, this could be easily extended with learning mechanisms (see Steingrube et al. (2010)) to switch to the desired gait when the respective behavioral scenarios are encountered, without any additional influence on the performance of the reservoir forward models.

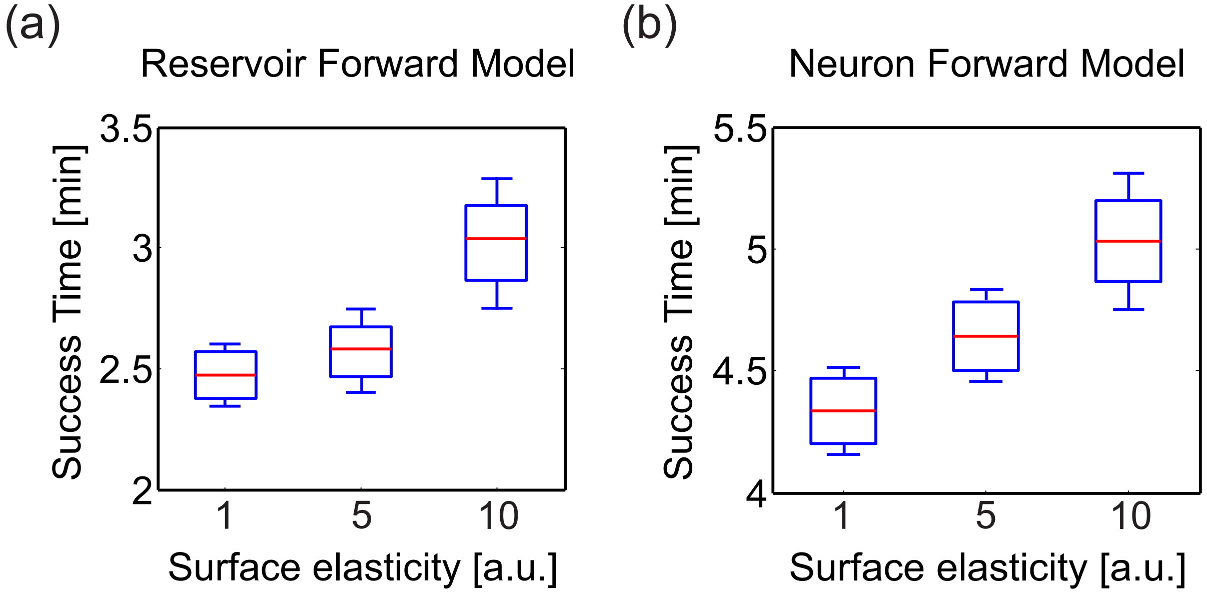


Figure 4.10: **Average time to succesfully overcome uneven terrains of different elasticity (hard, moderate, highly elastic)** (a) Average success time for reservoir-based forward model. (b) Average success time for adaptive neuron forward model from (Manoonpong et al., 2013b). Here the whiskers indicate one standard deviation above and below the mean value. Note the difference in scale of the y-axis in both plots.

In order to evaluate the performance of our adaptive reservoir forward model in comparison to the state of the art model recently presented in Manoonpong et al. (2013b) (single recurrent neural with low-pass filter), we carried out simulation experiments with AMOSII walking on different types of surfaces. Specifically, after training on a flat surface (under normal conditions) we carried out 10 trials each with the robot walking on uneven terrains (laid with multiple obstacles of height $8cm$), having three different elastic properties[4]. The surfaces were divided into hard (1.0), moderately elastic (5.0) and highly elastic (10.0). A tetrapod walking gait was used in all three cases. Starting from a fixed position, we noted the total time taken by the robot to successfully cross the uneven terrain region and move into a flat surface region. As observed in Figs. 4.10 (a) and (b), the reservoir forward model enables the robot to traverse the uneven region considerably faster as compared to the adaptive neuron forward model, in all three scenarios. Both the models can be seen to overcome the hard surface much better

---

[4] Here the elasticity coefficients do not strictly represent Young's modulus values. These were local parameter setting defined in the simulation, with increasing values causing greater elasticity.

as compared to the elastic ones. This was expected due to the changes in surface stiffness resulting in additional forces on the robot legs. However, the reservoir model performance was considerably more robust with a mean difference in success time of 1.86 mins for the hardest surface and approx. 2 mins for the most elastic surface, cases. Given that the walking gait was fixed, here the success time can be thought as an indicator of the robot's energy efficiency. In the absence of additional body mechanisms to deal with changing surface stiffness, the reservoir based model outperforms the previous implementations of adaptive forward models by $\approx 25\%$ order of magnitude on average.

4.5 Discussion

In this study, we presented adaptive forward models using the self-adaptive reservoir network for locomotion control. The model is implemented on each leg of a simulated bio-inspired hexapod robot. It is trained online during walking on a flat terrain in order to transform an efference copy (motor signal) into an expected foot contact signal (i.e., sensory prediction). Afterwards, the learned model of each leg is used to estimate walking states by comparing the expected foot contact signal with the actual incoming one. The difference between the expected and actual foot contact signals is used to adapt the robot's leg through elevation and searching control. Each leg is adapted independently. This enables the robot to successfully walk on uneven terrains. Moreover, using a backbone joint, the robot can also successfully cross a large gap and climb over a high obstacle as well as up a fleet of stairs. In this approach, basic walking patterns are generated by CPG-based control along with local leg reflex mechanisms that make use of the reservoir prediction to adapt the robot's behavior.

It is important to note that the usage of reservoir networks, as forward models here, provides the crucial benefit of an inherent representation of time and fading memory (due to the internal feedback loops and input dependent adaptations). Such memory of the time-varying motor or sensory stimuli is required to overcome intrinsic time lags between expected sensory signals and motor outputs (Wolpert et al., 1998), as well as in behavioral scenarios with considerable dependence on the history of motor output (Lonini et al., 2009). This is very difficult in most of the previous implementations of forward internal models using either simple single recurrent neuron implementations (Manoonpong et al., 2013b), feed-forward multi-layered neural networks (Schröder-Schetelig et al., 2010), or Bayesian network models (Dearden and Demiris, 2005), (Sturm et al., 2008). Furthermore, in this case, online adaptation of only the reservoir-to-readout weights (readout) makes such networks beneficial for simple and online learning.

The concept of forward models with efference copies in conjunction with neural control has been suggested since the mid-20th century (Holst and Mittelstaedt, 1950), (Held, 1961) and increasingly employed for biological investigations (Webb, 2004). This is because it can explain mechanisms which biological systems use to predict the consequence of their action based on sensory information, resulting in adaptive and robust behaviors in a closed-loop scenario. This concept also forms a major motivation for robots inspired by biological systems. Within this context, the work presented here, verifies that a combination of CPG-based neural control, adaptive reservoir forward models with efference copies, and searching and elevation control can be used for generating complex locomotion and adaptive behaviors in an artificial walking system. Additionally, although in this chapter we specifically focused on locomotive behaviors for walking robots, (such) SARN based motor prediction systems can be easily generalized to a number of other applications. Specifically for neuro-prosthetic (Ganguly and Carmena, 2009), sensor-driven orthotic control (Braun et al., 2014), (Lee and Lee, 2005) or brain-machine interface devices (Golub et al., 2012), that require the learning of such predictive models using highly non-stationary, temporal signals, applying SARN models can provide high performance gains, as compared to the current static feed-forward neural network solutions.

CHAPTER 5

NEUROMODULATORY COMBINED LEARNING OF GOAL-DIRECTED BEHAVIORS WITH RESERVOIR MODEL OF BASAL GANGLIA AND CORRELATION LEARNING MODEL OF CEREBELLUM

> ”A cat that once sat on a hot stove will never again sit on a hot stove. Or on a cold one either”.
>
> —*Mark Twain.*

In the previous chapters we have only considered synaptic plasticity in the form of supervised learning of the reservoir network weights. However biological systems are largely motivated by hedonistic returns. Typically this falls under the paradigm of reward-based learning, such that future actions are dependent on some function of the environmental feedback (rewards or punishments) and this guides the overall synaptic plasticity. Therefore, in this chapter, we demonstrate the usage of the self-adaptive reservoir network (SARN) model from within such a learning paradigm, where by, the reservoir synaptic connections can be modulated by external rewards (without the need of any supervised teacher signal). Furthermore, with a significant neuro-biological grounding, we motivate the possible neural correlate or brain structure (basal ganglia) that implements such reward modulated reservoir networks and works in combination with other brain areas (cerebellum), to guide goal-directed decision making. We also introduce a novel neuromodulatory rule for such a combined learning. Therefore, here we will spend considerable time exploring and motivating the underlying biological substrate of all these components. The over all goal of this chapter being, not just to demonstrate the usage of the SARN model from within a reward learning paradigm, but to demonstrate how such systems can be combined with other unsupervised learning mechanisms in the brain (namely correlation learning in cerebellum), to guide the overall goal-directed decision making in the brain (this forms a crucial part of temporal information processing in the timescale of few seconds to minutes, refer to Fig. 1.2).

Goal-directed decision making forms one of the key manifestation of closed loop temporal information processing. In biological systems, this is broadly based on associations between conditional and unconditional stimuli. This can be further classified as classical conditioning (correlation-based learning) and operant conditioning (reward-based learning). A number of computational and experimental studies have well established the role of the basal ganglia in reward-based learning, where as the cerebellum plays an important role in developing specific conditioned responses. Although viewed as distinct learning systems, recent animal experiments point towards their complementary role in behavioral learning, and also show the existence of substantial two-way communication between these two brain structures. Based on this notion of co-operative learning, here we hypothesize that the basal ganglia and cerebellar learning systems work in parallel and interact with each other. We envision that such an interaction is influenced by reward modulated heterosynaptic plasticity (RMHP) rule at the thalamus, guiding the overall goal directed behavior. Based on a number of recent experimental and theoretical studies showing high dimensional dynamics in the basal ganglia circuitry, here, we for the first time use a SARN based actor-critic model of the basal ganglia and a feed-forward correlation-based learning model of the cerebellum, whose learning outcomes can be combined (balanced) by a novel RMHP rule. This is tested using simulated environments of increasing complexity with a four-wheeled robot in a foraging task in both static and dynamic configurations. Although modeled with a simplified level of biological abstraction, we clearly demonstrate that a SARN based reward learning mechanism and correlation learning mechanism can be effectively combined by our RMHP rule, leading to stabler and faster learning of goal-directed behaviors, in comparison to the individual systems. Moreover, we also clearly demonstrate the need for such adaptive reservoir models in order to deal with scenarios having memory dependence of past sensory states or stimuli. In next few sections, we provide a computational model for adaptive combination of the basal ganglia and cerebellum learning systems by way of neuromodulated plasticity, that can lead to efficient goal-directed decision making in biological and bio-mimetic organisms.

5.1 Introduction

Associative learning by way of conditioning, forms the main behavioral paradigm that drives goal-directed decision making in biological organisms. Typically, this can be further classified into two classes, namely, classical conditioning (or correlation-based learning) (Pavlov, 1927) and operant conditioning (or reinforcement learning) (Skinner, 1938) . In general, classical conditioning is driven by associations between an early occurring conditional stimulus (CS) and a late occurring unconditional stimulus (US), which lead to conditioned responses (CR) or unconditioned responses (UR) in the organism (Freeman and Steinmetz, 2011), (Clark and Squire, 1998). The CS here acts as a predictor signal such that, after repeated pairing of the two stimuli, the behavior of the organism is driven by the CR (adaptive reflex action) at the occurrence of the predictive CS, much before the US arrives. The overall behavior is guided on the sole basis of stimulus-response (S-R) associations or correlations, without any explicit feedback in the form of rewards or punishments from the environment. In contrast

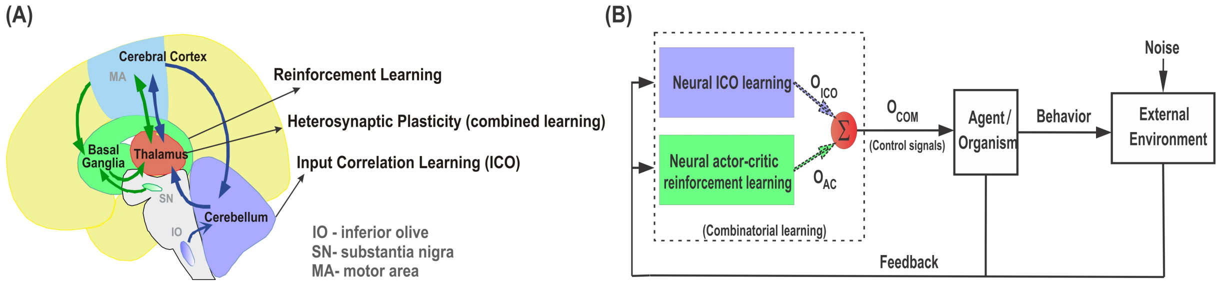


Figure 5.1: **(A)** Pictorial representation of the anatomical reciprocal connections between the basal ganglia, thalamus and cerebellum. Green arrows depict the cortico-striatal reward learning circuitry via the thalamus. Blue arrows depict the cortico-cerebellar recurrent loops for classically conditioned reflexive behaviors. Adapted and modified from (Doya, 2000a) **(B)** Combinatorial learning framework with parallel combination of ICO learning and actor-critic reinforcement learning. Individual learning mechanisms adapt their weights independently and then their final weighted outputs (O_{ico} and O_{ac}) are combined into O_{com} using a reward modulated heterosynaptic plasticity rule (dotted arrows represent plastic synapses). O_{com} controls the agent behavior (policy) while sensory feedback from the agent is sent back to both the learning mechanisms in parallel.

to such classically conditioned reflexive behavior acquisition, operant conditioning provides an organism with adaptive control over the environment with the help of explicit positive or negative reinforcements (evaluative feedback) given for corresponding actions. Over sufficient time, this enables the organism to respond with good behaviors, while avoiding bad or negative behaviors. As such within the computational learning framework, this is usually termed reinforcement learning (RL) (Sutton and Barto, 1998).

At a behavioral level, although the two conditioning paradigms of associative learning appear to be distinct from each other, they seem to occur in combination as suggested from several animal behavioral studies (Rescorla and Solomon, 1967), (Barnard, 2004), (Dayan and Balleine, 2002). Behavioral studies with rabbits (Lovibond, 1983) demonstrate that the strength of operant responses can be influenced by simultaneous presentation of classically conditioned stimuli. This was further elaborated upon in the behavior of fruit flies (Drosophila), where both classical and operant conditioning predictors influence the behavior at the same time and in turn improve the learned responses (Brembs and Heisenberg, 2000). On a neuronal level, this relates to the interaction between the reward modulated action selection at the basal ganglia and the correlation based delay conditioning at the cerebellum. Although the classical notion has been to regard the basal ganglia and the cerebellum to be primarily responsible for motor control, increasing evidence points towards their role in non-motor specific cognitive tasks like goal-directed decision making (Middleton and Strick, 1994),(Doya, 1999). Interestingly, recent experimental studies (Bostan et al., 2010), (Neychev et al., 2008) show that the the basal ganglia and cerebellum not only form multi-synaptic loops with the cerebral cortex, but, two-way communication between the structures exist via the thalamus Fig. 5.1 A) along with substantial disynaptic projections to the cerebellar cortex from the subthalamic nucleus (STN) of the basal ganglia and from the dentate nucleus (cerebellar output stage) to the striatum (basal ganglia input stage) (Hoshi et al., 2005). This suggests that the two structures are not separate performing distinct functional operations (Doya, 2000a), but are linked together forming an integrated functional network. Such integrated behavior is further illustrated in the timing and error prediction studies of (Dreher and Grafman, 2002) showing that the activation of the cerebellum and basal ganglia are not specific to switching attention, as previously believed, because both these regions were activated during switching between tasks as well as during the simultaneous maintenance of two tasks.

Based on these compelling evidences we formulate the *neural combined learning hypothesis*, which proposes that goal-directed decision making occurs with a parallel adaptive combination (balancing) of the two learning systems (Fig. 5.1 B) to guide the final action selection. As evident from experimental studies (Haber and Calzavara, 2009), the thalamus potentially plays a critical role in integrating the neural signals from the two sub-networks while having the ability to modulate behavior through dopaminergic projections from the ventral tagmental area (VTA)(Varela, 2014), (García-Cabezas et al., 2007). The motor thalamic (Mthal) relay nuclei, specifically the VA-VL (ventral anterior and ventral lateral) regions receive projections from the basal ganglia (inputs from the globas pallidus) as well as the cerebellum (inputs from the dentate nucleus) (Jones et al., 1985), (Percheron et al., 1996). This can be further segregated with the ventral anterior and the anterior region of the ventrolateral nucleus (VLa) receiving major inputs from the globus pallidus internus (GPi), while the posterior region of the ventrolateral nucleus (VLp) receives primary inputs from the cerebellum (Bosch-Bouju et al., 2013). Recent studies using molecular markers were able to distinguish the VA and VL nuclei in rats (Kuramoto et al., 2009), which had hitherto been difficult and were considered as a single overlapping area as the VA-VL complex. Interestingly, despite apparent anatomical segregation of information in the basal ganglia and cerebellar territories, similar ranges of firing rate and movement related activity are observed in the Mthal neurons across all regions (Anderson and Turner, 1991). Furthermore some experimental studies based on triple labeling techniques found zones of overlapping projections, as well as interdigitating foci of pallidal and cerebellar labels, particularly in border regions of the VLa (Sakai et al., 2000). In light of these evidences, it is plausible that the basal ganglia and cerebellar circuitries not only form an integrated functional network, but their individual outputs are combined together by a subset of the VLa neurons which in turn project to the supplementary and pre-supplementary motor cortical areas (Akkal et al., 2007) responsible for goal-directed movements. We envision that such a combined learning mechanism may be driven by reward modulated heterosynaptic plasticity (neuromodulation by way of dopaminergic projections) at the thalamus.

In this study, input correlation learning (ICO)in the form of a differential Hebbian learner (Porr and Wörgötter, 2006), was implemented as an example of delay conditioning in the cerebellum, while a self-adaptive reservoir network (Dasgupta et al., 2013a) (chapter 2) based continuous actor-critic learner (Doya, 2000b) was implemented as an example of reward based conditioning in the basal ganglia. Taking advantage of the individual learning mechanisms, the combined framework can learn the appropriate goal-directed control policy for an agent[1] in a fast and

---

[1] Agent here refers to any artificial or biological organism situated in a given environment.

robust manner outperforming the singular implementation of the individual components.

Although there have been a number of studies which have applied the two different conditioning concepts for studying self-organizing behavior in artificial agents and robots, they have mostly been applied separately to generate specific goal-directed behaviors (Prescott et al., 2006), (Morimoto and Doya, 2001), (Hofstoetter et al., 2002), (Manoonpong et al., 2007), (Verschure and Mintz, 2001). In our previous work (Manoonpong et al., 2013a) we motivated a combined approach of the two learning concepts on a purely algorithmic level without any adaptive combination between the two. To the best of our knowledge, in this paper we present for the first time a biologically plausible approach to model an adaptive combination of the cerebellar and basal ganglia learning systems, where they indirectly interact through sensory feedback . In this manner they work as a single functional unit to guide the behavior of artificial agents. Furhtermore, with the use of the reservoir model for basal-ganglia circuitry, we show that it clearly outperforms earlier feed-forward models for reward learning, specifically in decision making situations with dependence on memory of past sensory inputs. We test our neural combined learning hypothesis within the framework of goal-directed decision making using a simulated wheeled robot situated in environments of increasing complexity designed as part of static and dynamic foraging tasks (Sul et al., 2011). Our results clearly show that the proposed mechanism enables the artificial agent to successfully learn the task in the different environments with changing levels of interaction between the two learning systems. Although we take a simplified approach of simulated robot based goal-directed learning, we believe our model covers a reasonable level of biological abstraction that can help us understand better, the closed-loop interactions between these two neural subsystems as evident from experimental studies, display the use of reservoir based models from within the paradigm of reward learning (as opposite to the standard supervised principles) and also provide a computational model of such combined learning behavior which has hitherto been missing.

We now give a brief introduction to the neural substrates of the cerebellum and the basal ganglia with regards to classical and operant conditioning. Using a broad high-level view of the anatomical connections of these two brain structures, we motivate how goal-directed behavior is influenced by the respective structures and their associated neuronal connections. The individual computational models with implementation details of the two interacting learning systems are then presented in the Materials and methods section followed by results and discussion.

5.1.1 Classical Conditioning in the Cerebellum

The role of the Cerebellum and its associated circuitry in the acquisition and retention of anticipatory responses (sensory predictions) with Pavlovian delay conditioning has been well established (Christian and Thompson, 2003), (Thompson and Steinmetz, 2009). Although most of the classical conditioning studies are primarily based on eye-blink conditioning (Yeo and Hesslow, 1998), recent experimental studies have established the essential role of the cerebellum in learning and memory of goal-directed behavioral responses (Burguiere et al., 2010). In Fig. 5.2 A, a highly simplified control structure of the major cerebellar pathways and their relative function is indicated. The Inferior Olive relays the US signal to the cerebellar cortex through the climbing fibers and then induces plasticity at the synaptic junctions of the mossy fibers carrying the CS information (Herreros and Verschure, 2013). Repeated CS-US pairings gradually lead (through synaptic consolidation) to the acquisition of the CR with a drop in the firing activity of the Purkinje cells (output from the cerebellar cortex). The cerebral cortex projects to the lateral cerebellum via pontine nuclei relays (Proville et al., 2014), (Allen and Tsukahara, 1974), (Lisberger and Thach, 2013) which in turn have projections back to the cerebral cortex through relays in the thalamus (ventro-lateral nucleus), thus projecting the conditioned responses from the cerebellum to the motor cortical areas (Sakai et al., 2000), (Stepniewska et al., 1994). In essence, the cerebellar action modulates or controls the motor activity of the animal which produces changes in its goal oriented behavior. The goal oriented behaviors can typically involve both attraction towards or avoidance of specific actions (generally referred to as adaptive reflexes) involving both sensory predictions and motor control, towards which the cerebellum makes a major contribution. It is also important to note that although numerous experimental and computational studies demonstrate the function of the Cerebellum in classical conditioning or correlation learning (Kim and Thompson, 1997), (Woodruff-Pak and Disterhoft, 2008), a possible role of the Cerebellum towards supervised learning (SL) has also been widely suggested (Kawato et al., 2011), (Doya, 1999). Typically within the paradigm of SL a training or instructive signal acts as a reference towards which the output of a network (movements) is compared, such that the error generated acts as the driver signal to induce plasticity within the network in order to find the correct mapping between the sensory input stimuli and the desired outputs (Knudsen, 1994). Using the classical conditioning paradigm, it has been suggested that the instructive signal that supervises the learning is the input activity associated with the US. As such, the SL model of the cerebellum considers that the climbing fibers from the inferior olive provide the error signal (instructive activity) for the Purkinje cells. Coincident inputs from the inferior olive and the granule cells lead to plasticity at the granule-to-Purkinje synapses (Doya, 2000a). Although there have been experimental studies to validate the SL description of the cerebellum (Kitazawa et al., 1998), it has been largely directed towards considering the cerebellum as an internal model of the body and the environment (Kawato, 1999). Furthermore, (Krupa et al., 1993) observed that even when the red nucleus (relay between motor cortex and cerebellum) was inactivated learning proceeded with no CR being expressed. Thus, this demonstrates that no error signal based on the behavior was needed for learning to occur. Instead, the powerful climbing fiber activity evoked by the US, acting as a template, could cause the connection strengths of sensory inputs that are consistently correlated with it to increase. Subsequently, after sufficient repetition, the activity of these sensory inputs alone would drive the UR pathway. As such, in this work we directly consider correlation learning as the basis of classical conditioning in the cerebellum without taking into consideration SL mechanisms and do not explicitly consider the US relay from the inferior olive as an error signal.

5.1.2 Reward learning in the Basal Ganglia

In contrast to the role of the cerebellum in classical conditioning, the basal ganglia and its associated circuitry possess the necessary anatomical features (neural substrates) required for a

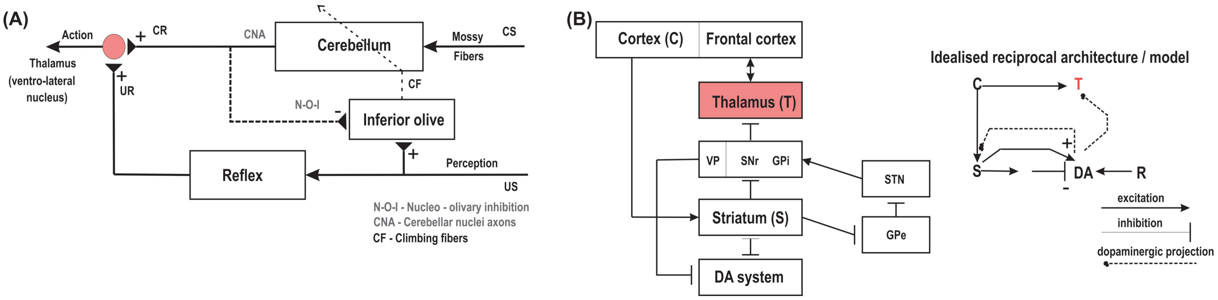


Figure 5.2: **(A)** Schema of the cerebellar controller with the reflexive pathways and anatomical projections leading the acquisition of reflexive behaviors. CS - conditioned stimulus, US - unconditioned stimuli, CR - conditioned response, UR - unconditioned response. **(B) (right)** Schematic representation of the neural architecture of the basal ganglia circuitry showing the layout of the various internal connections.**(left)** Shows the simplified circuit diagram with the main components as modeling in this paper using the reservoir actor-critic framework. Cortex = C, striatum = S, dopamine system = DA, reward = R, thalamus = T. Adapted and modified from (Wörgötter and Porr, 2005).

reward-based learning mechanism (Schultz and Dickinson, 2000). In Fig. 5.2 B, we depict the main anatomical connections of the cortical basal ganglia circuitry. It is comprised of the striatum (consisting of most of the caudate and the putamen, and of the nucleus accumbens), the internal (medial) and external (lateral) segments of the globus pallidus (GPi and GPe respectively), the subthalamic nucleus (STN), the ventral tegmental area (VTA) and the substantia nigra pars compacta (SNc) and pars reticulata (SNr). The input stage of the basal ganglia is the striatum connected via direct cortical projections. Previous studies have not only recognized the striatum as a critical structure in the learning of stimulus-response behaviors, but also established it as the major location which projects to as well as receives efferent connections from (via direct and indirect multi-synaptic pathways) the dopaminergic system (Joel and Weiner, 2000) (Kreitzer and Malenka, 2008). The processing of rewarding stimuli is primarily modulated by the dopamine neurons (DA system in Fig. 5.2 B) of the VTA and SNc with numerous experimental studies (Schultz and Dickinson, 2000) demonstrating, that changes in dopamine neurons encode the prediction error in appetitive learning scenarios, and associative learning in general (Puig and Mille, 2012). Fig. 5.2 B - right shows the idealized reciprocal architecture of the striatal and dopaminergic circuitry. Here sensory stimuli arrive as input from the cortex to the striatal network. Excitatory as well as inhibitory synapses project from the striatum to the DA system which in turn uses the changes in the activity of DA neurons to modulate the activity in the striatum. Such DA activity also acts as the neuromodulatory signal to the thalamus which receives indirect connections from the striatum, via the GPi, SNr and VTA (Varela, 2014). Computational modeling of such dopamine modulated reward learning behavior is particularly well reflected by the Temporal Difference (TD) algorithm (Sutton, 1988), (Suri and Schultz, 2001), as well as in the action selection based computational models of the basal ganglia (Gurney et al., 2001), (Humphries et al., 2006). In the context of basal ganglia modeling, Actor-Critic models (explained further in the next section) of TD learning (Houk et al., 1995), (Joel et al., 2002) have been extensively used. They create a functional separation between two sub-networks of the critic (modeling striatal and dopaminergic activity) and the actor (modeling striatal to motor thalamus projections). The TD learning rule uses the prediction error (TD error) between two subsequent predictions of the net weighted sum of future rewards based on current input and actions, to modulate critic weights via long-term synaptic plasticity. The same prediction error signal (dopaminergic projections) is also used to modulate the synaptic weights at the actor; output from which controls the the actions taken by the agent. Typically, here the mechanism of action selection, can be regarded as the neuromodulation process occurring at the striatum, which then reaches the motor thalamic regions via projections from the output stages of the basal ganglia, namely GPi/GPe and SNr (Gurney et al., 2001) (Houk et al., 2007) (Fig. 5.2 B).

5.2 Material & Methods

5.2.1 Combinatorial Learning with Reward Modulated Heterosynaptic Plasticity

According to the neural combined learning hypothesis for successful goal-directed decision making, the underlying neural machinery of animals combines basal ganglia and cerebellar learning systems output, induced with a reward modulated balancing (neuromodulation) between the two, at the thalamus to achieve net sensory-motor adaptation. Thus here we develop a system for the parallel combination of the input correlation-based learner (ICO) and the reward-based learner (actor-critic) as depicted in Fig. 5.1 B. The system works as a dual learner where the individual learning mechanisms run in parallel to guide the behavior of the agent. Both systems adapt their synaptic weights independently (as per their local synaptic modification rules) while receiving the same sensory feedback from the agent (environmental stimuli) in parallel. The final action that drives the agent is calculated as a weighted sum (Fig. 5.3 red circle) of the individual learning components. This can be described as follows:

$$o_{com}(t) = \xi_{ico} o_{ico}(t) + \xi_{ac} o_{ac}(t) \tag{5.1}$$

where, $o_{ico}(t)$ and $o_{ac}(t)$ are the t time step outputs of the input correlation-based learner and the actor-critic reinforcement learner, respectively. $o_{com}(t)$ represents the t time step combined action. The key parameters here that govern the learning behavior are the synaptic weights of the output neuron projection from the individual components, (ξ_{ico} and ξ_{ac}). These govern the degree of influence of the two learning systems, on the net action of the agent. Previously, a simple and straight forward approach was undertaken in (Manoonpong et al., 2013a), where an equal contribution ($\xi_{ico} = \xi_{ac} = 0.5$) of ICO and actor-critic RL for controlling an agent was considered. Although this can lead to successful solutions in certain goal-directed problems, it is sub-optimal due to the lack of any adaptive balancing mechanism. Intuitively for associative learning problems with immediate rewards the ICO system learns quickly as compared to distal

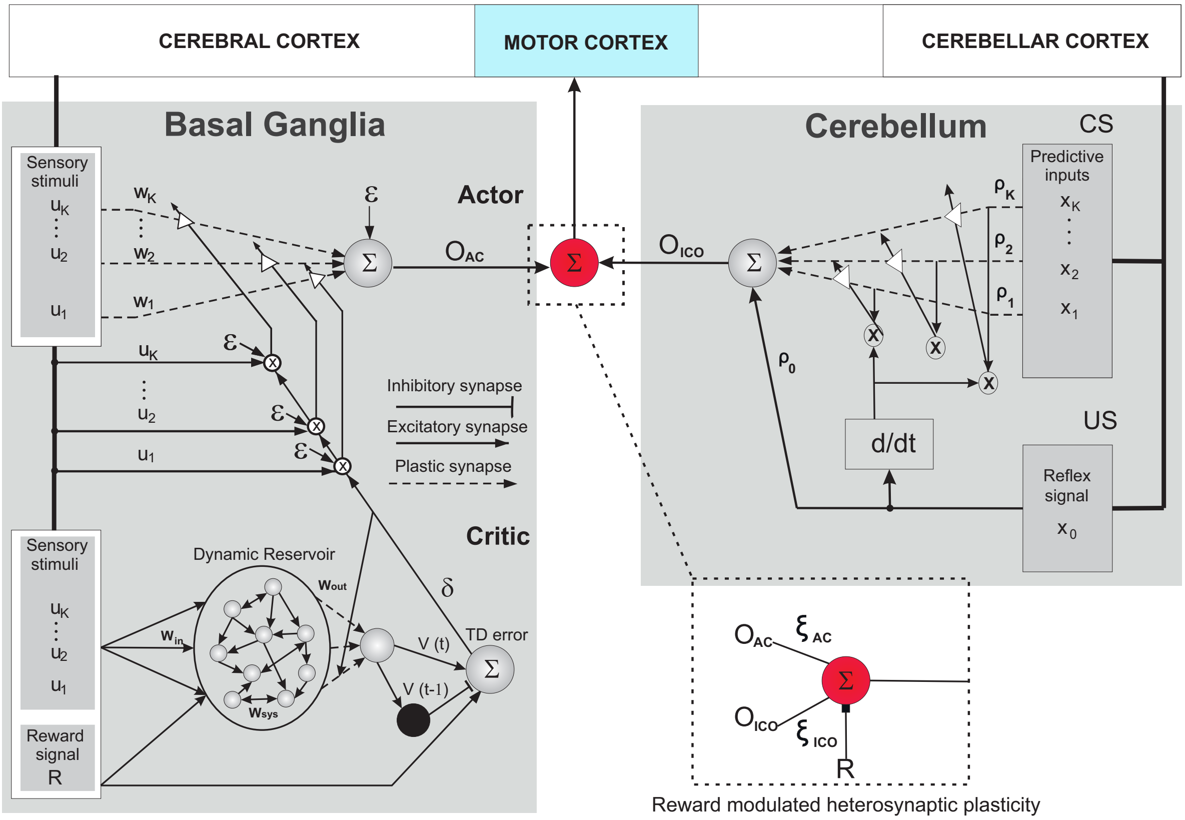


Figure 5.3: **Schematic wiring diagram of the combined learning neural circuit**: It consists of the reservoir actor-critic RL based on TD learning **(left)** and the input correlation learning (ICO) **(right)** models. The critic here is reminiscent of the cortico striatal connections modulated by dopaminergic neural activity (TD error). The actor represents the projections from the SNc, VTA and STN on to the thalamus where actions selection occurs. The ICO learning system is constructed in a manner similar to Fig. 5.1 C, with the inferior olive being represented by the differential Hebbian (d/dt) system that uses the US reflex signal to modulate the synaptic connections in the cerebellum. Explicit nucleo-olivary inhibitory connections were not modeled here. The red circle represents the communication junction which act as the integrator of the outputs from the two networks, being directly modulated by the reward signal R to control the overall action of the agent. (further details in text).

reward based goal-directed problems where, the ICO learner can provide guidance to the actor-critic learner. In particular depending on the type of problem, the right balance between the two learners needs to be achieved in an adaptive manner.
While there is evidence on the direct communication (Bostan et al., 2010) or combination of the subcortical loops from the cerebellum and the basal ganglia (Houk et al., 2007), a computational mechanism underlying this combination has not been presented, so far. Here we propose for the first time, an adaptive combination mechanism of the two components, modeled in the form of a reward modulated heterosynaptic plasticity (RMHP) rule, which learns the individual synaptic weights (ξ_{ico} and ξ_{ac}) for the projections from these two components. It is plausible that such a combination occurs at the VA-VL region of the motor thalamic nuclei which has both pallido-thalamic (basal ganglia) and cerebello-thalamic projections (Sakai et al., 2000). Furthermore a few previous experimental studies (Allen and Tsukahara, 1974), (Desiraju and Purpura, 1969) suggested that the individual neurons of the VL (nearly 20%) integrate signals from the basal ganglia and the cerebellum along with some weak cerebral inputs [2]. Based on biological evidence of dopaminergic projections at the thalamus from the basal ganglia circuit (Varela, 2014), (García-Cabezas et al., 2007) as well as cerebellar projections to the thalamic ventro-latral nucleus (Bosch-Bouju et al., 2013) (see Figure 42-7 in (Lisberger and Thach, 2013)) we consider here that such dopaminergic projections act as the neuromodulatory signal and triggers the heterosynaptic plasticity (Ishikawa et al., 2013). A large number of such heterosynaptic plasticity mechanisms contribute towards a variety of neural processes involving associative learning and development of neural circuits in general (Bailey et al., 2000) (Chistiakova and Volgushev, 2009). Although there is currently no direct experimental evidence of heterosynaptic plasticity at thalamic nuclei, it is highly plausible that such interactions could occur on synaptic afferents as observed in the amygdala and the hippocampus (Vitureira et al., 2012). Here, we use the instantaneous reward signal as the modulatory input in order to induce heterosynaptic changes at the thalamic junction. Similar approach have also been used in some previous theoretical models of reward modulated plasticity (Legenstein et al., 2008), (Hoerzer et al., 2012). Although the dopaminergic projections from the VTA to the Mthal are primarily believed to encode a reward prediction error (RPE) signal (Schultz and Dickinson, 2000), there exists considerable diversity in the VTA neuron types with a subset of these dopaminergic neurons directly responding to rewards (Cohen et al., 2012). Similar variability has also been observed in the single DA neuron recordings from memory guided sacadic tasks performed with primates (Takikawa et al., 2004). This suggests that although most dopaminergic neurons respond to a reward predicting conditional simuli, some may not strictly follow the canonical RPE coding (Cohen et al., 2012). It is important to note that, within this model, it is equally possible to use the reward prediction error (TD error, Eq. 5.10) and still learn the synaptic weights of the two components in a stable manner, however with a negligibly slower weight convergence due to continuous weight changes (see appendix A.4).

Based on this RMHP plasticity rule the ICO and actor-critic RL weights are learned at each

---

[2] It is also plausible that integration of activity arising in basal ganglia and cerebellum might take place in the thalamus nuclei other than the VL-VA, since pallidal as well as cerebellar fibers are known histologically to terminate not only in the VL-VA but also in other structures (Mehler, 1971)

time step as follows :

$$\Delta\xi_{ico}(t) = \eta R(t)[o_{ico}(t) - \bar{o}_{ico}(t)]o_{ac}(t), \tag{5.2}$$

$$\Delta\xi_{ac}(t) = \eta R(t)[o_{ac}(t) - \bar{o}_{ac}(t)]o_{ico}(t). \tag{5.3}$$

Here $R(t)$ is the current time step reward signal received by the agent, while $\bar{o}_{ico}(t)$ and $\bar{o}_{ac}(t)$ denote the low-pass filtered version of the output from the ICO learner and the actor-critic learner, respectively. They are calculated as:

$$\begin{aligned} \bar{o}_{ico}(t) &= 0.9\bar{o}_{ico}(t-1) + 0.1o_{ico}(t), \\ \bar{o}_{ac}(t) &= 0.9\bar{o}_{ac}(t-1) + 0.1o_{ac}(t). \end{aligned} \tag{5.4}$$

The plasticity model used here is based on the assumption that the net policy performance (agent's behavior) is influenced by a single global neuromodulatory signal. This relates to the dopaminergic projections to the ventra-lateral nucleus in the thalamus as well as connections from the amygdala which can carry reward related signals that influence over all action selection. The RMHP learning rule correlates three factors: 1) the reward signal, 2) the deviations of the ICO and actor-critic learner outputs from their mean values, and 3) the actual ICO and actor-critic outputs. The correlations are used to adjust their respective synaptic weights (ξ_{ico} and ξ_{ac}). Intuitively here the heterosynaptic plasticity rule can be also viewed as a homeostatic mechanism (Vitureira et al., 2012). Such that, the equation 2 tells the system to increase the ICO learners weights (ξ_{ico}) when the ICO output is coincident with the positive reward, while the third factor (o_{ac}) tells the system to increase ξ_{ico} more (or less) when the actor-critic learner weights (ξ_{ac}) are large (or small), and vice versa for equation 3. This ensures that overall the ratio of weight change of the two learning components occurs at largely the same rate. Additionally in order to prevent uncontrolled divergence in the learned weights, homeostatic synaptic normalization is carried out specifically as follows:

$$\begin{aligned} \xi_{ico}(t) &= \tfrac{\xi_{ico}(t)}{\xi_{ico}(t)+\xi_{ac}(t)}, \\ \xi_{ac}(t) &= \tfrac{\xi_{ac}(t)}{\xi_{ico}(t)+\xi_{ac}(t)}. \end{aligned} \tag{5.5}$$

This ensures that the synaptic weights always add up to one and $0 < \xi_{ico}, \xi_{ac} < 1$. In general this plasticity rule occurs on a very slow time scale which is governed by the learning rate parameter η. Typically convergence and stabilization of weights are achieved by setting η much

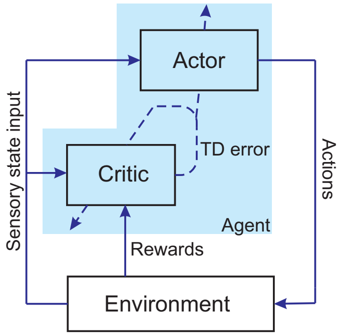


Figure 5.4: **Temporal difference actor-critic model of reward-based learning**

smaller compared to the learning rate of the two individual learning systems (ICO and actor-critic). To get a more detailed view of the implementation of the adaptive combinatorial learning mechanism, interested readers should refer to Algorithm 1, in appendix A.4 for the detailed algorithm.

5.2.2 Actor-critic Reservoir Model of Basal-ganglia Learning

TD learning (Sutton, 1988), (Suri and Schultz, 2001), in the framework of actor-critic reinforcement learning (Joel et al., 2002), (Wörgötter and Porr, 2005), is the most established computational model of the basal ganglia. As explained in the previous section, the TD learning technique is particularly well suited for replicating or understanding how reward related information is formed and transferred by the mid-brain dopaminergic activity.

The model consists of two sub-networks, namely, the adaptive critic and the actor (Fig. 5.4). The critic is adaptive in the sense that it learns to predict the weighted sum of future rewards taking into account the current incoming time varying sensory stimuli and the actions (behaviors) performed by the agent within a particular environment. The difference between the predicted ”value” of sum of future rewards and the actual measure acts as the temporal difference (TD) prediction error signal that provides an evaluative feedback (or reinforcement signal) to drive the actor, as well as modulate the predictions by the critic. Eventually the actor learns to perform the proper set of actions (policy[3]) that maximize the weighted sum of future rewards as computed by the critic. The evaluative feedback (TD error signal) in general acts as a measure of goodness of behavior that, overtime, lets the agent learn to anticipate reinforcing events. Within this computational framework, the TD prediction error signal and learning at the critic are analogous to the dopaminergic (DA) activity and the DA dependent long term synaptic plasticity in the striatum (Fig. 5.2 B), while the remaining parts of striatal circuitry can be envisioned as the actor which uses the TD modulated activity to generate actions, which

---

[3] In reinforcement learning, policy refers to the set of actions performed by an agent that maximizes it’s average future reward.

drives the agent's behavior.

Based on the reservoir computing framework (Maass et al., 2002), (Jaeger and Haas, 2004), here we demonstrate the use of the self-adaptive reservoir network (SARN) (Dasgupta et al., 2013a) as the adaptive critic (cortico-striatal circuitry and the DA system) mechanism (Fig. 5.3 left below). This is connected to a feed-forward neural network, serving the purpose of the part of striatum that performs action selection (Gurney et al., 2001) and then relays it to the motor thalamus via projections from the globus pallidus and substantia nigra. Given the ability of SARN to inherently represent temporal information of incoming stimuli, this provides a novel framework to model a continuous actor-critic reinforcement learning scheme, which is particularly suited for goal-directed learning in continuous state-action problems, while at the same time maintaining a reasonable level of biological abstraction (Fremaux et al., 2013). Here, the reservoir network can be envisioned as analogous to the cortex and its inherent recurrent connectivity structure, and the readout neurons serving as the striatum, with plastic projections from the recurrent layer, as the modifiable cortico-striatal connections (Hinaut and Dominey, 2013). The reservoir network is constructed as a generic network model of N recurrently connected neurons with high sparsity (refer to Tab. A.1 in appendix A.4 for details) and fixed synaptic connectivity. The connections within the recurrent layer are drawn randomly in order to generate a sparsely connected network of inhibitory and excitatory synapses. A subset of the reservoir neurons receive input connections (fixed synaptic strengths) as external driving signals and has an additional output layer of neurons that learns to produce a desired response based on synaptic modification of weights from the reservoir to output neurons. The input connections along with the large recurrently connected reservoir network represents the main cortical microcircuit-to-striatum connections, while the output layer neural activity can be envisioned as striatal neuronal responses.

In this case, the reservoir critic provides an input (sensory stimuli) driven dynamic network with a large repertoire of signals that is used to predict the value function v (average sum of future rewards). $v(t)$ approximates the accumulated sum of the future rewards $R(t)$ with a given discount factor γ $(0 \leq \gamma < 1)$[4] as follows:

$$v(t) = \sum_{i=1}^{\infty} \gamma^{i-1} R(t+i). \tag{5.6}$$

All the remaining components in the reservoir network is the same as presented before in chapters 2 and 3. However now, rather than use a predefined supervised target signal to modulate the reservoir-to-readout weights $\mathbf{W}^{out}$, we make use of the TD error signal generated based on reservoir predictions. Here, the membrane potential at the soma (at time t) of the reservoir neurons, resulting from the incoming excitatory and inhibitory synaptic inputs, is given by the N dimensional vector of neuron state activation's, $\mathbf{x}(t) = x_1(t), x_2(t), ..., x_N(t)$. The input to the reservoir network, consisting of the agent's states (sensory input stimuli from the cerebral

---

[4] The discount factor helps assigning decreasing value to rewards further away in the past as compared to the current reward.

cortex), is represented by the K dimensional vector $\mathbf{u}(t) = u_1(t), u_2(t), ..., u_K(t)$. The recurrent neural activity within the dynamic reservoir varies as a function of its previous state activation and the current driving input stimuli. Recollect, that the recurrent network dynamics is given by,

$$x_i(t+1) = (1 - \tfrac{\Delta t}{\tau_i})x_i(t) + \tfrac{\Delta t}{\tau_i}\Big(g \textstyle\sum_{j=1}^{N} W_{i,j}^{rec} r_j(t) + \sum_{j=1}^{K} W_{i,j}^{in} u_j(t) + B_i\Big), \tag{5.7}$$

$$\hat{v}(t) = z(t) = \mathtt{tanh}(\mathbf{W}^{out}\mathbf{r}(t)), \tag{5.8}$$

$$r_i(t) = \mathtt{tanh}(a_i x_i(t) + b_i). \tag{5.9}$$

Where, all the parameters are the same as in the basic SARN model, with the exception of the readout neuron activity $\hat{v}(t) = z(t)$. Here, instead of the a linear function, the readout neuron output is also calculated with a tan hyperbolic non-linear transfer function.

Based on the TD learning principle, the primary goal of the reservoir critic is to predict $v(t)$ such that the TD error δ is minimized over time. At each time point t, δ is computed from the current ($\hat{v}(t)$) and previous ($\hat{v}(t - \Delta t)$) value function predictions (reservoir output), and the current reward signal $R(t)$, as follows:

$$\delta(t) = R(t) + \gamma\hat{v}(t) - \hat{v}(t - \Delta t). \tag{5.10}$$

The readout weights $\mathbf{W}^{out}$ are calculated using the recursive least squares (RLS) formulation (section 2.2.4, Eq. 2.32) at each time step, while the sensory stimuli $\mathbf{u}(t)$ are being fed into the reservoir. Unlike the supervised learning formulation, here the error signal for weight modulation was not calculated based on a target output, but $\mathbf{W}^{out}$ were adapted, such that the overall TD-error (δ - here acts as the instantaneous error signal) is minimized. The readout weight update is defined as:

$$\mathbf{W}^{out}(t) = \mathbf{W}^{out}(t - \Delta t) - \delta(t)\mathbf{P}(t)\mathbf{r}(t) \tag{5.11}$$

where, $\mathbf{P}$ is a $N \times N$ square matrix proportional to the inverse of the correlation matrix of reservoir neuron firing rate vector $\mathbf{r}$. As depicted in Eq. 2.33 and Eq. 2.34, it was initialized with a small constant parameter δ_c, and updated at each time point as,

$$\mathbf{P}(t) = \mathbf{P}(t - \Delta t) - \left(\frac{\mathbf{P}(t - \Delta t)\mathbf{r}(t)\mathbf{r}^T(t)\mathbf{p}(t - \Delta t)}{1 + \mathbf{r}^T(t)\mathbf{P}(t - \Delta t)\mathbf{r}(t)} \right). \tag{5.12}$$

As introduced previously, (chapter 2) (Dasgupta et al., 2013a) generic intrinsic plasticity mechanism (Eq. 2.28 and Eq. 2.29) based on the Weibull distribution for unsupervised adaptation of the reservoir neuron non-linearity using a stochastic decent algorithm to adapt the scale a_i and shape parameters b_i of the reservoir neuron non-linearity was carried out as pre-training process. This was coupled with the adaptation of individual neuron timeconstants τ_i (Eq. 2.19) based on the incoming sensory state information. It is also important to note that one of the primary assumptions of the basic TD learning rule is a Markovian one, which considers future sensory cues and rewards depending only on the current sensory cue without any memory component. The use of a reservoir critic (due to the inherent fading temporal memory) breaks this assumption. As a result, such design principle extends our model to generic decision making problems with short term dependence of immediate sensory stimuli on the preceding history of stimuli (agents states) and reward (see Fig. 5.5 for a simulated example of local temporal memory in reservoir neurons, elaborate examples can be seen in chapter 3). This was not possible in traditional models of an adaptive critic based on feed-forward radial-basis function (RBF) networks (Doya, 2000a), and as such is another crucial contribution of the reward learning formulation of SARN.

The actor (Fig. 5.3 left above) is designed as a single stochastic neuron, such that for a one dimensional action generation the output (O_{ac}) is given as:

$$o_{ac}(t) = \epsilon(t) + \sum_{i=1}^{K} w_i(t)u_i(t), \tag{5.13}$$

where K denotes the dimension (total number) of sensory stimuli ($\mathbf{u}(t)$) to the agent being controlled. The parameter w_i denotes the synaptic weights for the different sensory inputs projecting to the actor neuron. Stochastic noise is added to the actor via $\epsilon(t)$, which is the exploration quantity updated at every time step. This acts as a noise term, such that initially exploration is high, and the agent needs to navigate the environment more if the expected cumulative future reward $v(t)$ is sub-optimal. However, as the agent learns to successfully predict the maximum cumulative reward (value function) over time, and the net exploration is decreased. As a result $\epsilon(t)$ gradually tends towards zero as the agent starts to learn the desired behavior (correct policy). Using Gaussian white noise σ (zero mean and standard deviation one) bounded by the minimum and maximum limits of the value function (v_{min} and v_{max}), the exploration term is modulated as follows:

$$\epsilon(t) = \Omega\sigma(t) \cdot \mathtt{min} \left[0.5, \mathtt{max} \left(0, \frac{v_{max} - \hat{v}(t)}{v_{max} - v_{min}} \right) \right]. \tag{5.14}$$

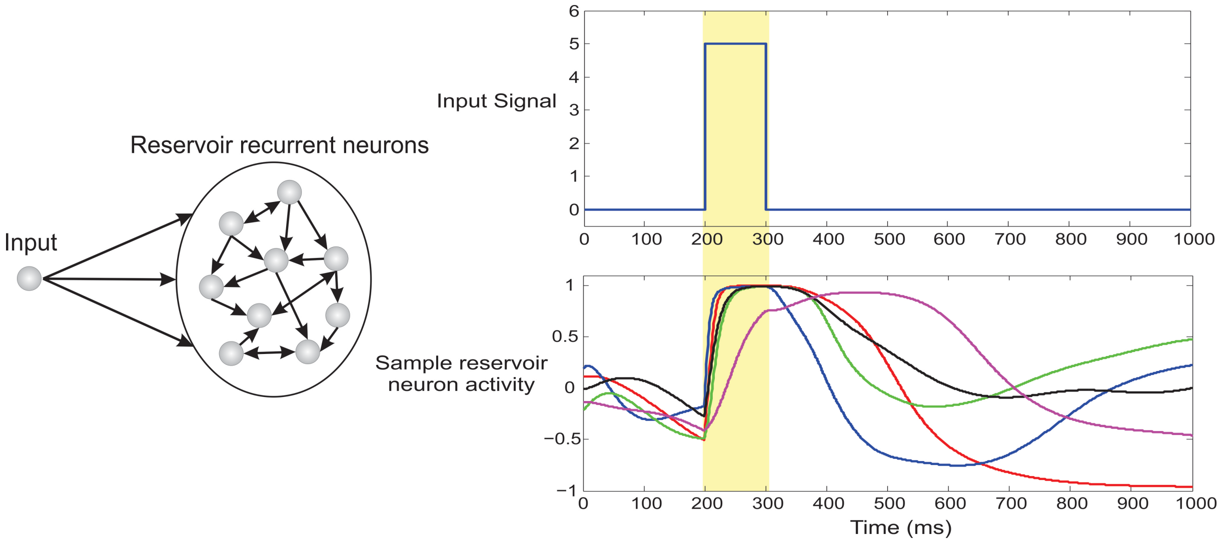


Figure 5.5: **Fading temporal memory in recurrent neurons of dynamic reservoir** The recurrent network (100 neurons) was driven by a brief 100 ms pulse and a fixed auxiliary input of magnitude 0.3 (not shown here). Spontaneous dynamics then unfolds in the system based on Eq. 5.7. The lower right panel plots the activity of 5 randomly selected recurrent neurons. It can be clearly observed that the driving input signal clamps the activity of the network at 200 ms however different neurons decay with varying timescale. As a result the network exhibits considerable fading memory of the brief incoming input stimuli.

Here, Ω is a constant scale factor selected empirically (see appendix for details). The actor learns to produce the correct policy, by an online adaptation (Fig. 5.3 left above) of its synaptic weights w_i at each time step as follows:

$$\Delta w_i(t) = \tau_a \delta(t) u_i(t) \epsilon(t), \tag{5.15}$$

where τ_a is the learning rate such that $0 < \tau_a < 1$. Instead of using direct reward $r(t)$ to update the input to actor neuron synaptic weights, using the TD-error (i.e. error of an internal reward) allows the agent to learn successful behavior, even in cases of delayed reward scenarios (reward is not given uniformly for each time step but is delivered as a constant value after a set of actions were performed to reach a specific goal). In general, once the agent learns the correct behavior, the exploration term ($\epsilon(t)$) should become zero, as a result of which no further weight change (Eq. 5.15) occurs and $o_{ac}(t)$ represents the desired action policy, without any additional noise component.

5.2.3 Input Correlation Model of Cerebellar Learning

In order to model classical conditioning of adaptive motor reflexes[5] in the cerebellum, we use a model-free, correlation based, predictive control learning rule called input correlation learning (ICO) (Porr and Wörgötter, 2006). ICO learning provides a fast and stable mechanism in order to acquire and generate sensory predictions for adaptive responses based solely on the correlations between incoming stimuli. The ICO learning rule (Fig. 5.3 Right) takes the form of an unsupervised synaptic modification mechanism using the cross-correlation between the incoming predictive input stimuli (predictive here means that the signals occur early) and a single reflex signal (late occurring). As depicted in Fig. 5.3 right, cortical perceptual input in the form of predictive signals (CS) represents the mossy fiber projections to the cerebellum microcircuit, while the Climbing fiber projections from the inferior olive that modulates the synaptic weights in the deep cerebellar nucleus are depicted in a simplified form with the differential region (d/dt).

The goal of the ICO mechanism is to behave as a forward model system (Porr and Wörgötter, 2006) that uses the sensory CS to predict the occurrence of the innate reflex signal (external predefined feedback signaling unwanted scenarios), thus letting the agent to react in an anticipatory manner to avoid the basic reflex altogether. Based on a differential Hebbian learning rule (Kolodziejski et al., 2008) the synaptic weights in the ICO scheme are modified using heterosynaptic interactions of the incoming inputs, depending on their order of occurrence. In general, the plastic synapses of the predictive inputs get strengthened if they precede the reflex signal and are weakened if their order of occurrence is reversed. As a result, the ICO learning rule

---

[5] The reflex signal is typically a default response to an unwanted situation. This acts as the unconditional stimulus occurring later in time, than the predictive conditional stimulus.

drives the behavior depending on the timing of correlated neural signals. This can be formally represented as,

$$o_{ico}(t) = \rho_0 x_0(t) + \sum_{j=1}^{K} \rho_j(t) x_j(t). \tag{5.16}$$

Here, o_{ico} represents the output neuron activation of the ICO system driven by the superposition of the plastic K-dimensional predictive inputs $x_j(t) = x_1(t), x_2(t), ..., x_K(t)$[6] (differentially modified) and the fixed innate reflex signal $x_0(t)$. The synaptic strength of the reflex signal is represented by ρ_0 and is fixed to the constant value of 1.0 in order to signal innate response to the agent. Using the cross-correlations between the input signals, our differential Hebbian learning rule modifies synaptic connections as follows:

$$\Delta\rho_j(t) = \mu x_j(t) \frac{d}{dt} x_0(t). \tag{5.17}$$

Here, μ defines the learning rate and is typically set to a small value to allow slow growth of synaptic weights with convergence occurring once the reflex signal $x_o = 0$ (Porr and Wörgötter, 2006). Thus ICO learning allows the agent to predict the primary reflex and successfully generate early, adaptive actions. However no explicit feedback of goodness of behavior is provided to the agent and thus only an anticipatory response can be learned without the explicit notion of how well the action allows reaching a desired (rewarding) goal location. As depicted in Fig. 5.3, the output from the ICO learner is directly fed into the RMHP unit envisioned to be part of the ventro-lateral thalamic nucleus (Bosch-Bouju et al., 2013), (Akkal et al., 2007).

5.3 Results

In order to test the performance of our bio-inspired adaptive combinatorial learning mechanism, and validate the interaction through sensory feedback, between reservoir model of reward learning (basal ganglia) and correlation-based learning (cerebellum) systems, we employ a simulated, goal-directed decision making scenario of foraging behavior. This is carried out within a simplified paradigm of a four-wheeled robot navigating an enclosed environment, with gradually increasing task complexity.

---

[6] This x(t) is different from the neural state activation vector $\mathbf{x}(t)$ of equation 9.

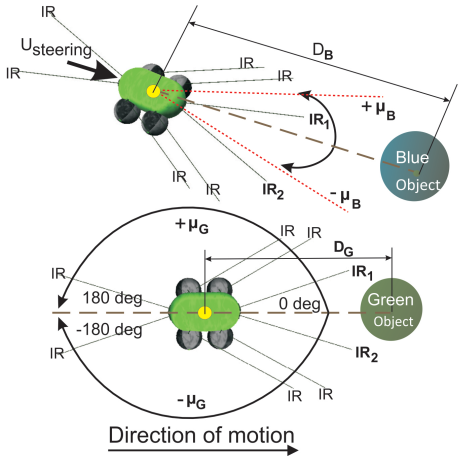


Figure 5.6: **Simulated mobile robot system for goal-directed behavior task**. **(Top)** The mobile robot NIMM4 with different types of sensors. The relative orientation sensor μ is used as state information for the robot. **(Bottom)** Variation of the relative orientation μ_G to the green goal. the front left and right infrared sensors IR_1 and IR_2 are used to detect obstacles in front of the robot. Direction control for the robot is maintained using the quantity $U_{steering}$ calculated by the individual learning components (ICO and actor-critic) and then fed to the robot wheels to generate forward motion or steering behavior. Sensors D_G and D_B measure straight line distance to the goal locations.

5.3.1 Robot model

The simulated wheeled robot NIMM4 (Fig. 5.6) consists of a simple body design with four wheels whose collective degree of rotation controls the steering and the over all direction of motion. It is provided with two front infrared sensors (IR_1 and IR_2) which can be used to detect obstacles to its left or right side, respectively. Two relative orientation sensors (μ_G and μ_B) are also provided, which can continuously measure the angle of deviation of the robot with respect to the green (positive) and blue (negative) food sources. They are calibrated to take values in the interval $[-180^o, 180^o]$ with the angle of deviation $\mu_{G,B} = 0^o$ when the respective goal is directly in front of the robot, $\mu_{G,B}$ is positive when the goal locations are to the right of the robot and negative for the opposite case. In addition NIMM4 also consists of two relative position sensors ($D_{G,B}$) that can calculate it's relative straight line distance to a goal, taking values in the interval $[0, 1]$, with the respective sensor reading tending to zero, as the robot gets closer to the goal location and vice versa.

5.3.2 Experimental setup

The experimental setup (Fig. 5.7) consists of a bounded environment with two different food sources (desired vs punishing) located at fixed positions. The primary task of the robot is to navigate the environment such that, eventually, it should learn to steer towards the food source

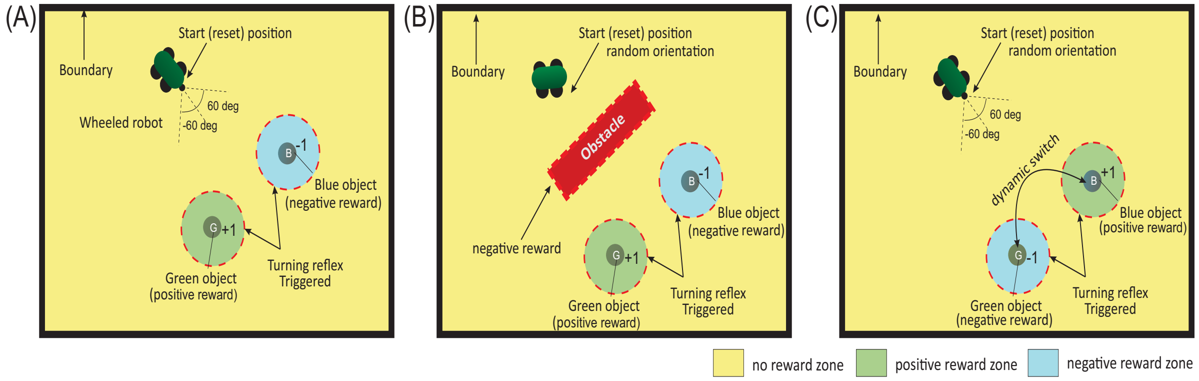


Figure 5.7: **Three different scenarios for the goal-directed foraging task**. **(A)** Environmental setup without an obstacle case. Green and Blue objects represent the two food sources with positive and negative rewards, respectively. The red dotted circle indicates the region where the turning reflex response (from the ICO learner) kicks in. The robot is started from and reset to the same position, with random orientation at the beginning of each trial episode. **(B)** Environmental setup with an obstacle. In addition to the previous setup, a large obstacle is place in the middle of the environment. The robot needs to learn to successfully avoid it and reach the rewarding food source. Collisions with the obstacle (triggered by IR_1 and IR_2) generate negative rewards (-1 signal) to the robot. **(C)** Environmental setup with dynamic switching of the two objects. It is an extended version of the first scenario. After every 50 trials the reward zones are switched such that the robot has to dynamically adjust to the new positively reinforced location (food) and learn a new trajectory from the starting location.

that leads to positive reinforcements (green spherical ball in Figs. 5.7 A, B, and C) while avoiding the goal location that provides negative reinforcements or punishments (blue spherical ball), within a specific time interval. The main task is designed as a continuous state-action problem with a distal reward setup (Reinforcement zone in Fig. 5.7), such that the robot starts at a fixed spatial location with random initial orientation ($[-60°, 60°]$) and receives the positive or negative reinforcement signal only within a radius of specific distance ($D_{G,B} = 0.2$) from the two goal locations. Within this boundary, for the green goal it receives a continuous reward of +1 at every time step and a continuous punishment of -1 in case of the blue goal, respectively. At other locations along the environment no reinforcement signal is given to the robot.

The experiments are further divided into three different scenarios of, foraging without an obstacle (case I), with single obstacle (case II) and a dynamic foraging scenario (case III) demonstrating different degrees of reward modulated adaptation between the two learning systems in different environments. In all three basic scenarios, the robot can continuously sense its angle of deviation to the two goals with $\mu_{G,B}$ always active. This acts as a Markov decision process (MDP) (Sutton and Barto, 1998) such that, the next sensory state of the robot depends on the sensory information for the current state of the robot and the action it performed, and is conditionally independent of all the previous sensory states and actions. In order to test the influence of the reservoir based model of critic, on tasks requiring memory of past sensory inputs, a variation of case I was also carried out with the environment designed to be partially observable (Fig. 5.11). Such that, the robot cannot sense its direction (angle of deviation $\mu_{G,B}$ inactive) to either of the goals until it reaches half way distance to either of them, i.e. $D_{G,B} = 0.6$. This makes this scenario, a partially observable Markov decision problem (POMDP) (Spaan, 2012) that requires memory of past sensory states and actions in order to calculate the next state and take the appropriate action. In all cases, detecting the obstacle results in negative reinforcement (continuous -1 signal) triggered by the front infrared sensors ($IR_{1,2} > 1.0$). Furthermore, hitting the boundary wall in the arena results in a negative reinforcement signal (-1), with the robot being reset to the original starting location. Although the robot is provided with relative distance sensors, sensory stimuli (state information) is provided using only the angle of deviation sensors and the infrared sensors. The reinforcement zone (distance of $D_{G,B} = 0.2$) is also used as the zone of reflex to trigger a reflex signal for the ICO learner. 50 runs were carried out for each setup in all cases. Each run consisted of a maximum of 200 trials. The robot was reset if the maximum simulation time of 15s was reached, or if it reaches one of the goal locations or if it hits a boundary wall, which ever occurs earlier.

5.3.3 Cerebellar system: ICO learning setup

The cerebellar system in the form of ICO learning (Fig. 5.3 right) was setup as follows: $\mu_{G,B}$ were used as predictive signals (CS). Two independent reflex signals ($x_{0,B}$ and $x_{0,G}$, see Eq. 5.16) were configured with one for blue food source and the other for the green food source (US). The setup was designed following the principles of delayed conditioning experiments, where, an overlap between the CS and the US stimuli needs to exist in order for the learning to take place. The

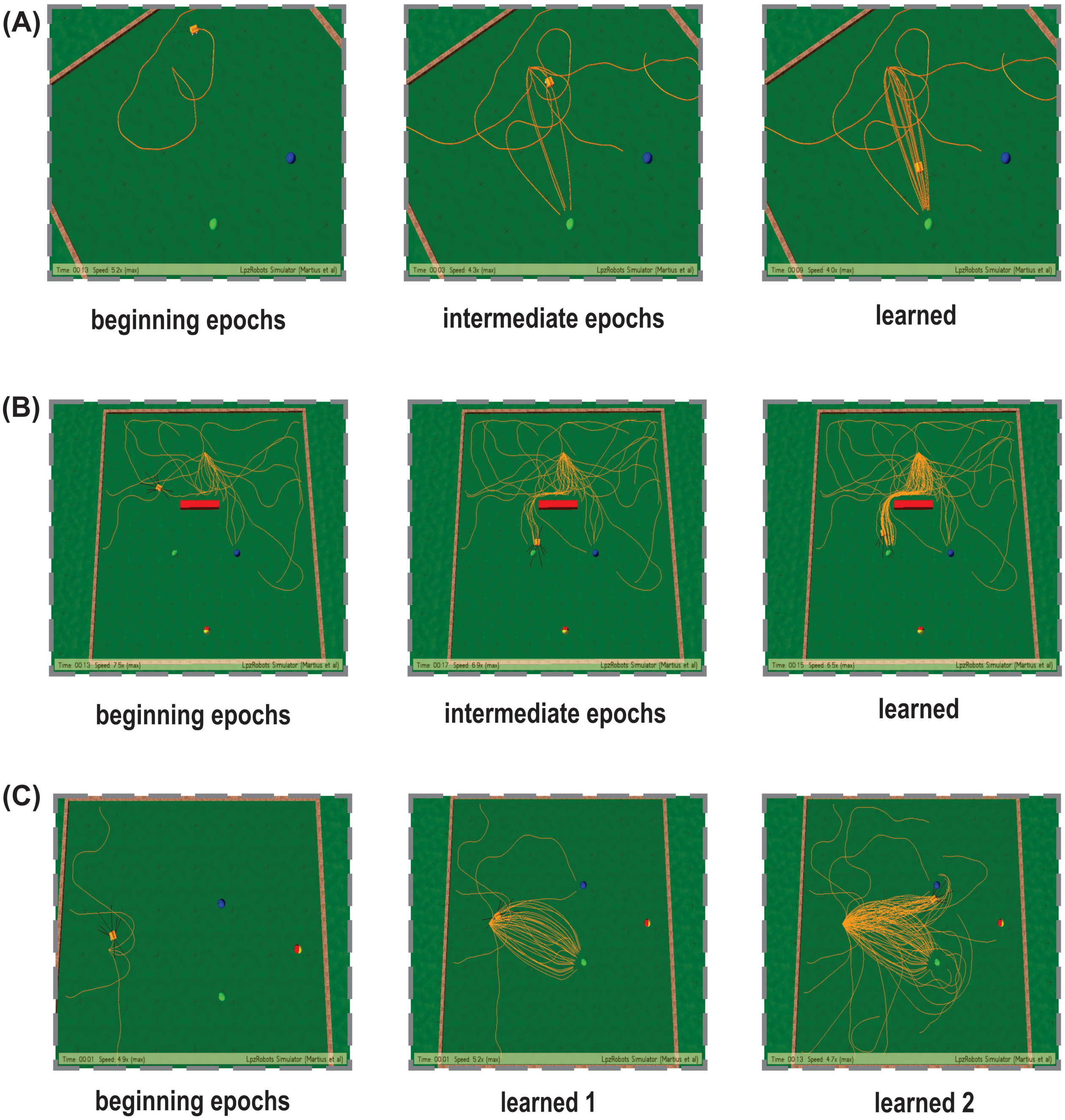


Figure 5.8: **Simulation snapshots of the robot learning for the three cases taken at specific epochs of time**. **(A)** Snapshots of the learning behavior for the static foraging task without obstacles. **(B)** Snapshots of the learning behavior for the static foraging task with a single obstacle. **(C)** Snapshots of the learning behavior for the dynamic foraging task. Panel learned 1 - represents the learned behavior for the initial task of reaching the green goal. After 50 trials, the reward stimulus was changes and the new desired (positively reinforced) location was the blue goal. Panel learned - 2 represents the learned behavior after dynamic switching of reward signals. *Please see electronic version for better resolution.*

reflex signal was designed (measured in terms of the relative orientation sensors of the robot) to elicit a turn towards a specific goal once the robot comes within the reflex zone (inside the dotted circle in Fig. 5.7 and Fig. 5.11 A). Irrespective of the kind of goal (desired or undesired) the reflex signal drives the robot towards it with a turn proportional to the deviations defined by $\mu_{G,B}$ i.e large deviations cause sharper turns. The green and the blue ball were placed such that there was no overlap between the reflex areas, hence only one reflex signal per goal, got triggered at a time. In other words, the goal of the ICO learner is simply to learn to steer towards a food location without any knowledge of it's worth. This is representative of an adaptive reflexive behavior as observed in rodent foraging studies where in the behavior is guided without explicit rewards, but just driven by conditioning between the CS-US stimuli, such that the robot or animal learns to favor certain spots in the environments without any knowledge of their worth. The weights of the ICO learner ρ_{μ_G} and ρ_{μ_B} (Eq. 5.16) with respect to the green and blue goals were initialized to 0.0. If the positive derivative of the reflex signal becomes greater than a predefined threshold, the weights change and otherwise they remain static, i.e a higher change in ρ_{μ_G} in comparison to ρ_{μ_B} would mean that the robot gets drawn towards the green goal more.

5.3.4 Basal ganglia system: Reservoir Actor-critic setup

The basal ganglia system in the form of the self-adaptive reservoir based actor-critic learner was setup such that, the inputs to the critic and actor networks (Fig. 5.3 left) consisted of the two relative orientation sensor data μ_G and μ_B and the front left and right infrared sensors (IR_1 and IR_2) of the robot (Fig. 5.6). Although the robot also contains relative distance sensors, these were not used as state information inputs. This makes the task less trivial, such that sufficient but not complete information was provided to the actor-critic RL network. The reservoir network for the critic consisted of $N = 100$ neurons and one output neuron that estimates the value function $v(t)$ (Eq. 5.8). Reservoir input weights W^{in} were drawn from an uniform distribution $[-0.5, 0.5]$ while the reservoir recurrent weights W^{rec} were drawn from a Gaussian distribution of mean 0 and standard deviation g^2/N (see Eq. 5.7). Here g acts as the scaling factor for W^{rec}, and it was designed such that there is only 10% internal connectivity in W^{rec} with a scaling factor of 1.2. The reward signal $R(t)$ (Eq. 5.10) was set to +1 when the robot comes close (reflex/reinforcement zone) to the green ball and to -1 when it comes close to the blue ball. A negative reward of -1 was also given for any collisions with the boundary walls or obstacle. At all other locations within the environment, the robot receives no explicit reward signal. Thus the setup is designed keeping a delayed reward scenario in mind, such that earlier actions lead to a positive or negative reward, only when the robot enters the respective reinforcement/reflex zone. The synaptic weights of the actor with respect to the two orientation sensors (w_{μ_G} and w_{μ_B}) were initialized to 0.0, while the weights with respect to the infrared sensors (w_{IR_1} and w_{IR_2}) were initialized to 0.5 (Eq. 5.13). After learning, a high value of w_{μ_G} and a low value of w_{μ_B} would drive the robot towards the green goal location and away from the blue goal. The weights of the infrared sensor inputs effectively control the turning behavior of the robot when encountered with an obstacle (higher w_{IR_1} - right turn, higher w_{IR_2} - left turn). The parameters of the adaptive combinatorial network are summarized in the appendix A.4.

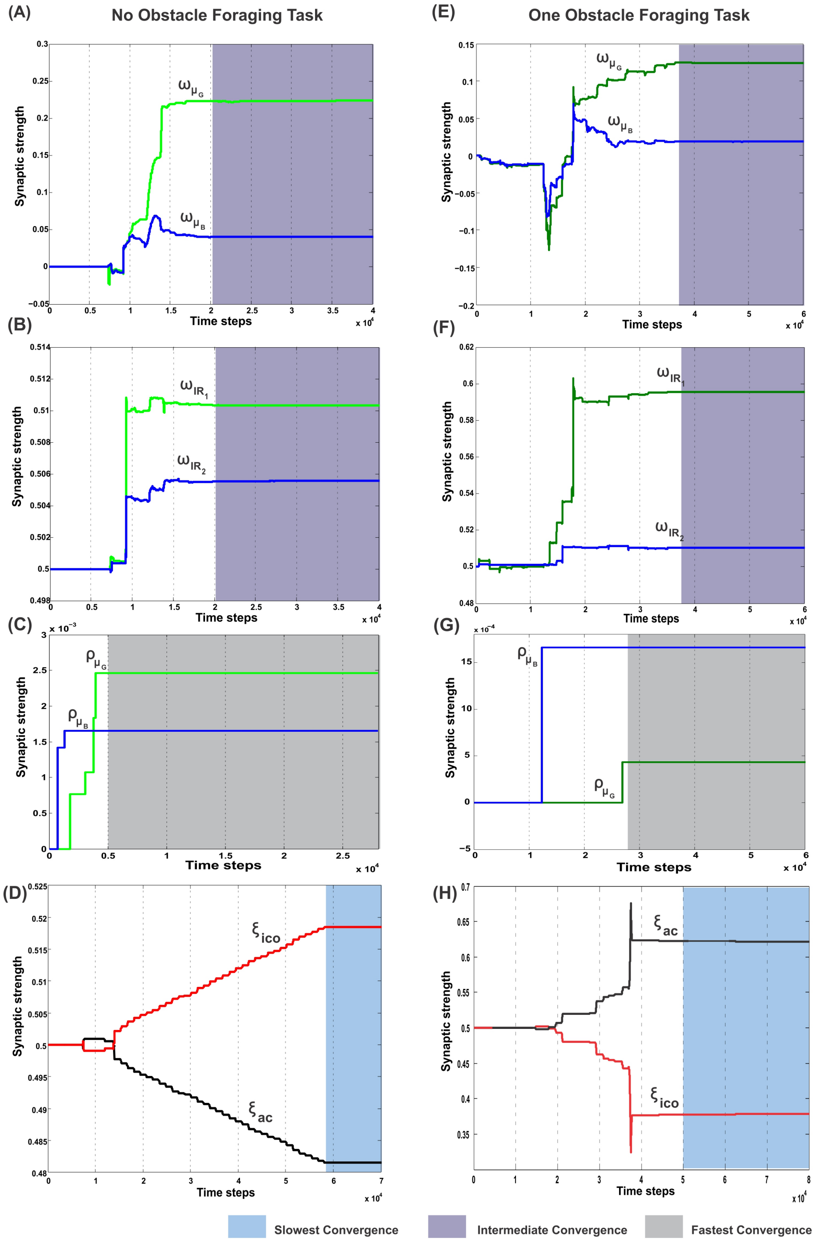


Figure 5.9: **Synaptic weight change curves for the static foraging tasks without obstacle and with single obstacle .** **(A)** Change in the synaptic weights for actor-critic RL learner. Here $w_{\mu G}$ corresponds to the input weights of the orientation sensor towards the green goal and $w_{\mu B}$ corresponds to the input weights of the orientation sensor towards the blue goal. **(B)** Change in the weights of the two infrared sensor inputs of the actor. w_{IR_1} is the left IR sensor weight, w_{IR_2} is the right IR sensor weights. **(C)** Change in the synaptic weights of the ICO learner. $\rho_{\mu G}$ is the CS stimulus weight for the orientation sensor towards green, $\rho_{\mu B}$ the CS stimulus weight for the orientation sensor towards blue. **(D)** Learning curve of the RMHP combined learning mechanism showing the change in the weights of the individual components. ξ_{ico} is weight of the ICO network output. ξ_{ac} is weight of the actor-critic RL network output. **(E)-(H)** Show the change in the weights corresponding to the single obstacle static foraging task. In all the plots the grey shaded region marks the region of convergence for the respective synaptic weights. Three different timescales exist in the system, with the ICO learning being the fastest, actor-critic RL being intermediate and the adaptive combined learning being the slowest (see text for more details.)

5.3.5 Case I: Foraging without obstacle

In the simplest foraging scenario the robot was placed in an environment with two possible food sources (green and blue) and without any obstacle in between (Fig. 5.7 A). In this case the green food source provided positive reward while the blue food source provided negative reward. The goal of the combined learning mechanism was to make the robot successfully steer towards the desired food source. Fig. 5.8 A, shows simulation snapshots of the behavior of the robot as it explores the environment. As observed from the trajectory of the robot, initially it performed a lot of exploratory behavior and randomly moved around in the environment, but eventually it learned to move solely towards the green goal. This can be further analyzed looking at the development of the synaptic weights of the different learning components as depicted in Fig. 5.9. As observed in Fig. 5.9 C, due to the simple correlation mechanism of the ICO learner (cerebellar system), the ICO weights adapt relatively faster as compared to the actor. Due to random explorations (Fig. 5.10 B) in the beginning, in the event of the blue goal being visited more frequently, reflexive pull towards blue goal - ρ_{μ_B} is greater than towards the green goal - ρ_{μ_G}. However, after sufficient explorations, as the robot starts reaching the green goal more frequently, ρ_{μ_G} also starts developing. This is counteracted by the actor weights (basal ganglia system), where in, there is a higher increase in w_{μ_G} (orientation sensor input representing angle of deviation from green goal) as compared to w_{μ_B} (orientation sensor input representing angle of deviation from blue goal). This is caused as result of the increased positive rewards received from the green goal (Fig. 5.10 A) that causes the TD-error to modulate the actor weights (Eq. 5.15) accordingly. At the same time no significant change is seen in the infrared sensor input weights (Fig. 5.9 B), due to the fact that in this scenario, the infrared sensors get triggered only on collisions with the boundary wall and remain dormant otherwise. Recall that the infrared sensor weights were initialized to 0.5.

Over time as the robot moves more towards the desired food source, the ICO weights also stabilize with the reflex towards the green goal being much stronger. This also leads to a reduction of the exploration noise (Fig. 5.10 B), and the actor weights eventually converge to a stable value (Figs. 5.9 A and 5.9 B). Here, the slow RMHP rule performs a balancing act between the two learning systems with initial higher weight of the actor-critic learner and then a switch towards the ICO system, once the individual learning rules have converged. Fig. 5.9 C shows the development of the value function ($v(t)$) at each trial, as estimated by the critic. As observed initially the critic underestimates the total value due to high explorations and random navigation in the environment. However as the different learning rules converge, the value function starts to reflect the total accumulated reward with stabilization after 25 trials (each trials consisted of approximately 1000 time steps).

This is also clearly observed from the change of the orientation sensor readings shown in Fig. 5.10 D. Although there is considerable change in the sensor readings initially, after learning, the orientation sensor towards the green goal (μ_G) records positive angle, while the orientation from the blue goal μ_B records considerably lower negative angles. This indicates that the robot learns to move stably towards the positively rewarded food source and away from the oppositely rewarded blue food source. Although this is the simplest foraging scenario, the development of the RMHP weights ξ_{ico} and ξ_{ac} (Fig. 5.9 D) depicts the adaptive combination of the basal

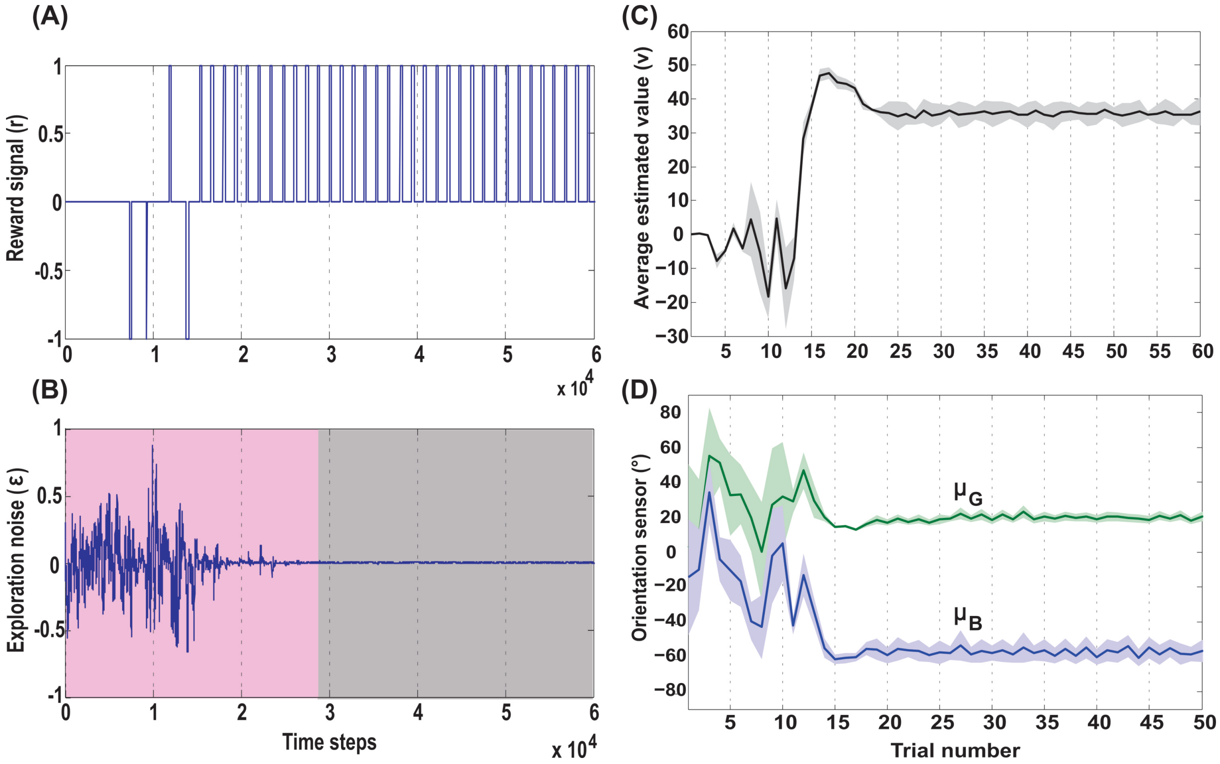


Figure 5.10: **Temporal development of key parameters of the actor-critic RL network, in the no obstacle foraging task**. **(A)** Development of the reward signal (r) over time. Initially the robot receives a mix of positive and negative rewards due to random explorations. Upon successfully learning the task, the robot is steered towards the green goal every time, receiving only positive rewards. **(B)** Development of the exploration noise (ϵ) for the actor. During learning there is a high noise in the system (pink shaded region), which causes the the synaptic weights of the actor to change continuously. Once the robot starts reaching the green goal more often the TD error from the critic decreases leading to a decrease in exploration noise (grey shaded region), which in turn causes the weights to stabilize (Fig. 5.8). **(C)** Average estimated value (v) as predicted by the reservoir critic is plotted for each trial. The maximum estimated value is reached after about 18 trials after which the exploration steadily decreases and the value function prediction also reaches near convergence at 25 trials (1 trial approximates 1000 time steps). The thick black line represents the average value calculated over 50 runs of the experiment with standard deviation given by the shaded region. **(D)** Plots of the two orientation sensor readings (in degrees) for the green (μ_G) and the blue (μ_B) goals, averaged over 50 runs. During initial exploration the angle of the deviation of the robot from the two goals changes randomly. However after convergence of the learning rules, the orientation sensor readings stabilize with small positive angle of deviation towards the green goal and large negative deviation from the blue goal. This shows that post learning, the robot steers more towards the green goal and away from the blue goal. Here the thick lines represent average values and the shaded regions represent standard deviation.

gangliar and cerebellar learning systems for goal-directed behavior control. Here the cerebellar system (namely ICO) acts as a fast adaptive reflex learner that guides and shapes the behavior of the reward-based learning system. Although both the individual systems eventually converge to provide the correct weights towards the green goal, the higher strength of the ICO component (ξ_{ico}) leads to a good trajectory irrespective of the starting orientation of the robot. This is further illustrated in the simulation video showing three different scenarios of only ICO, only actor-critic and the combined learning cases, see simulation video at http://manoonpong.com/Nimm4/Video1NoObstacle.mp4.

Special Scenario Case I: Partially Observable

In order to test the performance of our reservoir model of actor-critic learning (basal ganglia) in generic scenarios requiring temporal memory of past sensory states (input stimuli to critic and actor), we modified the case I environment to be partially observable (Fig. 5.11 A) (Dasgupta et al., 2013b). Specifically, starting from the same fixed location (with random orientations), the robot was sensory deprived (unable to measure its angle of deviations from either of the goals) until it reaches halfway to the goal locations. In other words, unlike the previous scenario, now the robot could sense its orientation with respect to the two goals only within a certain range ($D_{G,B} < 0.6$) and not in every location in the environment. This makes the environment, in classical terms, partially observable. As such, the future states and actions of the robot are dependent on not only its immediately previous state, but also on the history of past states (memory). Furthermore, keeping the ICO learning component fixed (cerebellum), we tested both the previous scenario and the modified case I using our reservoir actor-critic model as well as standard adaptive feedforward radial-basis function (RBF) networks (Manoonpong et al., 2013a), (Morimoto and Doya, 1998). Here the RBF actor-critic network (see Manoonpong et al. (2013a) for details) was constructed such that the critic network size varies between 15 to 100 hidden RBF units.

The actor-critic learner was setup as follows: The inputs to the critic and actor networks (Fig. 2) consisted of the two relative orientation sensor data ϕ_G and ϕ_B. The reservoir network for the critic consisted of $N = 100$ neurons and one ouput neuron that estimates the value function $v(t)$ (Eq. (1)). Reservoir input weights W_{in} were drawn from an uniform distribution $[-0.5, 0.5]$ while the reservoir recurrent weights W_{sys} were drawn from the uniform distribution $[-1, 1]$. W_{sys} was subsequently scaled to a spectral radius of 0.9 with only 10% internal connectivity. The reward signal $r(t)$ (Eq. (2)) was set to $+1$ when the robot comes close to the green ball and to -1 when it comes close to the blue ball. A RBF feedforward network was used for comparison with the reservoir based critic. The RBF critic size was varied from 16 to 100 hidden neurons.

The performance of the reservoir based critic as compared to the RBF critic (keeping all other components of the combinatorial learning mechanism the same) is compared in Fig. 5.11 B, with respect to the fully and the partially observable scenarios of the same task. As observed, the reservoir based critic clearly outperforms the RBF critic. Moreover the difference in performance is highly significant in the POMDP scenario, where the reservoir network outperforms the RBF critic by a success rate greater than 50%. Temporal memory of incoming agent state information

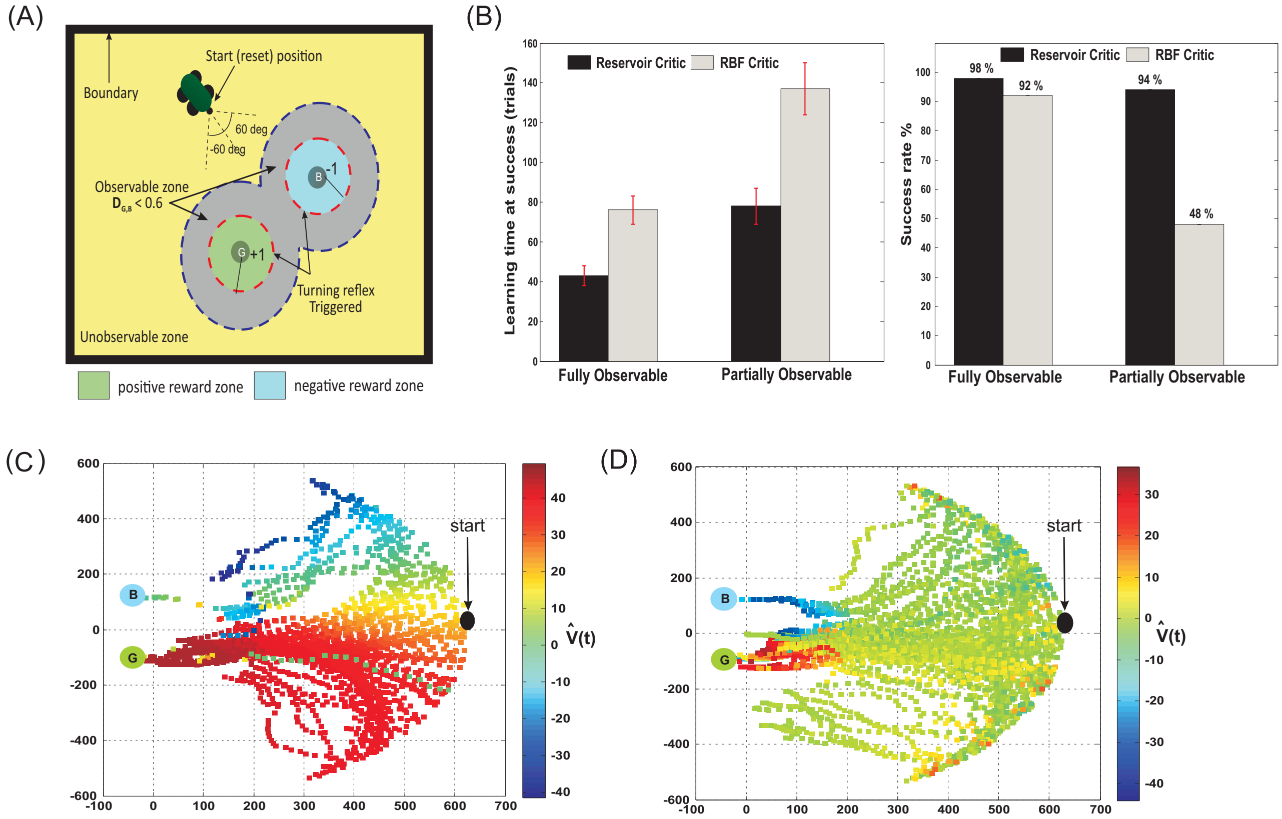


Figure 5.11: **Performance comparison between a reservoir based critic and RBF based critic for the fully observable and partialy observable cases (ICO and actor components remained the same)** . **(A)** Environmental setup for the partially observable case. The robot can sense its relative orientation to the goals only when within the observable zone (filled grey dotted circles). Reinforcement is received similar to the fully observable case. Due to this sensory deprivation between the starting location and beginning of the observable zone, the robot depends on its history of previous state information to guide its future behavior, making the problem a POMDP. **(B)** Left - average learning time (trials) needed to succesfully complete the task, calculated over 50 experiments (error bars indicate standard deviation for 95 % confidence interval), Right - Success rate in percentage. Here ”success” indicates the robots ability to correctly navigate to the green goal. **(C)** Estimation of the value function $v(t)$ using reservoir based critic. The $\hat{v}(t)$ estimate is plotted with respect to local co-ordinates of the robot and an observer located directly opposite to the robot starting position. Colormap indicates the changing $\hat{v}(t)$ values. The black ball indicates the starting position of the robot with random orientation and the curvature of the plot is resultant of the shape of view from the observer. **(D)** Estimation of value function $v(t)$ for the same task using the static RBF based critic.

available to the reservoir critic is crucial for solving complex non-markovian decision making problem, as compared to memoryless feedforward critic networks. Furthermore although both the implementations have almost similar success rate for the fully observable case, the reservoir based system converges to a solution (learned behavior of driving the robot to the green goal) significantly faster (less than 50 trials), as observed in Fig. 5.11 B right. However, expectantly the POMDP scenario takes longer time to learn the correct behavior, owing to the reduction in the total sensory information available to the system. Upon successfully learning the task the weights of the actor (Eq. 5.15) converge such that the robot gets pulled towards the desired green goal. It should be noted that although linear stochastic actors were used in this setup, the POMDP scenario is effectively solved due to the inherent trace of previous inputs in the reservoir critic. In contrast the memoryless RBF critic system works on chance and hence learns the POMDP task with less than 50% success rate. To elaborate this further, in Figs. 5.11 C and D, we compare the performance of the reservoir based critic with a RBF critic network in terms of the value function estimation curves for the same goal-directed behavior task (i.e. case I without obstacles). It is clearly observed that the reservoir critic successfully enables the mobile robot to learn to drive towards the green goal while avoiding the blue goal. Furthermore unlike the RBF critic (Fig. 5.11 D), the value function curve in Figs. 5.11 C, displays a strong gradient of the estimated value of $\hat{v}(t)$ with high positive values towards the correct goal (green object). In contrast the memory less RBF critic estimates $\hat{v}(t)$ to values closer to zero in most locations except for regions within the zone of reward. As a result our adaptive reservoir critic learns the task faster as indicated by the fast convergence (time to success) in Fig. 5.11 B right.

Having established the efficiency of the reservoir network actor-critic model, in the next cases we will only consider the fully observable case, and with changes in complexity of the environment, evaluate the performance of the individual learning components and their combination with the novel RMHP rule.

5.3.6 Case II: Foraging with single obstacle

In order to evaluate the efficacy of the two learning systems and their cooperative behavior, the robot was now placed in a slightly modified environment (Fig 5.7 B). As in the previous case, the robot still starts from a fixed location with initial random orientations. However, it now has to overcome an obstacle placed directly in front (field of view), in order to reach the rewarding food source (green goal). Collisions with the obstacle, during learning, resulted in negative rewards (-1) triggered by the front left (IR_1) and right (IR_2) infrared sensors. This influenced the actor-critic learner to modulate the actor weights via TD-error and generate turning behavior around the obstacles. In parallel, the ICO system, still learns only a default reflexive behavior of getting attracted towards either of the food sources by a magnitude proportional to its proximity to them (same as case I), irrespective of the associated rewards. As observed from the simulation snapshots in Fig 5.8 B, after initial random exploration, the robot learns the correct trajectory to navigate around the obstacle and reach the green goal. From the synaptic weight development curves for the actor neuron (Fig. 5.9 E) it is clearly observed that although initially there is a competition between w_{μ_G} and w_{μ_B}, after sufficient exploration, as the robot gets more positive

rewards by moving to the green food source, the w_{μ_G} weight becomes larger in magnitude and eventually stabilizes.
Concurrently in Fig. 5.9 F, it can be observed that unlike the previous case the left infrared sensor input weight w_{IR_1} gets considerably higher as compared to w_{IR_2}. This is indicative of the robot learning the correct behavior of turning right in order to avoid the obstacle and reach the green goal. However interestingly, as opposed to the simple case (no obstacle) the ICO learner tries to pull the robot more towards the blue goal, as seen from the weight development of ρ_{μ_G} and ρ_{μ_B} in Fig. 5.9 G. This behavior can be attributed to the fact that, as the robot reaches the blue object in the beginning, the fast ICO learner provides high weights for a reflexive pull towards the blue as opposed to the green goal. As learning proceeds and the robot learns to move towards the desired location (driven by the actor-critic system), the ρ_{μ_G} weight also increases, however it still continues to favor the blue goal. As a result in order to learn the correct behavior the combined learning systems needs to favor the actor-critic mechanism more as compared to the naive reflexives from the ICO. This is clearly observed from the balancing between the two as depicted in the ξ_{ico} and ξ_{ac} weights in Fig. 5.9 H. Following the stabilization of the individual learning system weights, the combined learner provides much higher weighting of the actor-critic RL system. Thus in this scenario, due to the added complexity of an obstacle, one sees that the reward modulated plasticity (RMHP rule) learns to balance the two interacting learning systems, such that the robot still performs the correct decisions overtime (see the simulation video at, http://manoonpong.com/Nimm4/Video2SingleObstacleStatic.mp4).

5.3.7 Case III: Dynamic foraging (reversal learning)

A number of modeling as well as experimental studies of decision making (Sugrue et al., 2004) have considered the behavioral effects of associative learning mechanisms on dynamic foraging tasks as compared to static ones. Thus, in order to test the robustness of our learning model, we changed the original setup (Fig. 5.7 C), such that, initially a positive reward (+1) is given for the green object and a negative reward (-1) for the blue one. This enables the robot to learn moving towards the green object while avoiding the blue object. However after every 50 trials the sign of the rewards was switched such that now the blue object received positive reward, and the green goal the opposite. As a result the learning system needs to quickly adapt to the new situation and learn to navigate to the correct target. As observed in the Fig. 5.12 B, initially the robot performs random explorations receiving a mixture of positive and negative rewards, however after sufficient trials, the robot reaches a stable configuration (exploration drops to zero) and receives positive rewards concurrently (Fig. 5.12 A). This corresponds to the previous case of learning to move towards the green goal. As the rewards were switched, the robot then obtained negative reward when it moved to the green object. As a consequence, the exploration gradually increased again; thereby the robot also exhibited random movements. After successive trials, a new stable configuration was reached with the exploration dropping to zero and now the robot received more positive rewards, however for the other target (blue object). This is depicted with more clarity, in the simulation snapshots in Fig. 5.8 C (beginning - random explorations, learn 1 - reaching green goal, learn 2 - reaching blue goal).

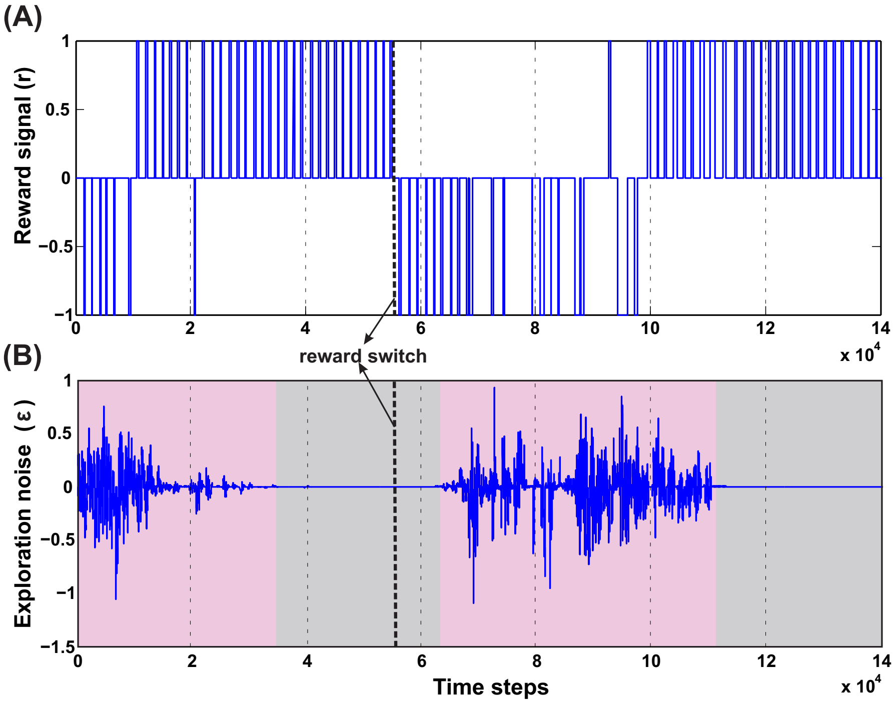


Figure 5.12: **Temporal development of the reward and exploration noise for the dynamic foraging task**. **(A)** Change in the reward signal (r) over time. Between 3×10^4 time steps and 5×10^4 time steps the robot learns the initial task of reaching the green goal, receiving positive rewards (+1), successively. However after 50 trials (approximately 5×10^4 to 5.5×10^4 time steps) the reward signals were changed, causing the robot to receive negative rewards (-1) as it drives to the green goal. After around 10×10^4 time steps as the robot learns to steer correctly towards the new desired location (blue goal), it successively receives positive rewards. **(B)** Change in the exploration noise (ϵ) over time. There is random exploration in the beginning of the task and after switching the reward signals (pink shaded regions), followed by stabilization and decrease in exploratory noise once the robot learns the correct behavior (gray shaded region). In both plots the thick dashed line (black) marks the point of reward switch.

In order to understand how the combined learning mechanism handles this dynamic switching, in Fig. 5.13 we plot the synaptic weight developments of the different components.

Initially the robot behavior is shaped by the ICO weights (Fig. 5.13 B) which learn to steer the robot to the desired location, such that the reflex towards green object (ρ_{μ_G}) is stronger than that towards the blue object (ρ_{μ_B}). Furthermore as the robot receives more positive rewards, the basal ganglia system starts influencing it's behavior by steadily increasing the actor weights towards the green object (Fig. 5.13 A, $w_{\mu_G}, w_{IR_1} > w_{\mu_B}, w_{IR_2}$). This eventually causes the exploration noise (ϵ) to decrease to zero and the robot learns a stable trajectory towards the desired food source. This corresponds to the initial stable region of the synaptic weights between $2X10^4$ and $6X10^4$ time steps in Figs. 5.13 A, B and C. Interestingly the adaptive RMHP rule tries to balance the influence from the two learning systems with eventual higher weighting of the ICO learner. This is similar to the behavior observed in the no obstacle static scenario (Fig. 5.9 D). After 50 trials ($5X10^4$ time steps), the reward signs were inverted which causes the exploration noise to increase. As a result the synaptic weights try to adapt once again and influence the behavior of the robot,now towards the blue object. In this scenario although the actor weights eventually converge to the correct configuration of w_{μ_B} greater than w_{μ_G}, the cerebellar reflexive behavior remains biased towards the green object (previously learned stable trajectory). This can be explained from the fact that the cerebellar or ICO learner has no knowledge of the type of reinforcement received from the food sources, and just naively tries to attract the robot to a goal when it is close enough (within the zone of reflex) to it. As a result of this behavior, the RMHP rule tries to balance the contributions of both learning mechanisms, by increasing the strength of the actor-critic RL component as compared to the ICO learner component ($\xi_{ac} > \xi_{ico}$). This lets the robot, now learn the opposite behavior of stable navigation towards the blue food source, causing the exploration noise to decrease once again. Thus through the adaptive combination of the different learning systems, modulated by the RMHP mechanism, the robot was able to deal with dynamic changes in environment and complete the foraging task successfully (see the simulation video at, http://manoonpong.com/Nimm4/Video3Dynamic.mp4).

Furthermore as observed from the rate of success on the dynamic foraging task (Fig. 5.14 A), the RMHP based adaptive combinatorial learning mechanism clearly outperforms the individual systems (only ICO or only actor-critic RL). Here the rate of success was calculated as the percentage of times the robot was able to successfully complete the first task of learning to reach the green food source (green colored bars), and then after switching of the rewards signals, the percentage of times it successfully reached the blue food source (blue colored bars). Furthermore in order to test the influence of the RMHP rule, we tested the combined learning with both, equal weightage to ICO and actor-critic systems as well as a plasticity induced weighting for the two individual learning components. It was observed that although for the initial static case of learning to reach the green goal the combined learning mechanism with equal weights works well, the performance drops considerably, after the reward signals were switched, and re-adaptation was required. Such a performance was also observed in our previous work (Manoonpong et al., 2013a) using a simple combined learning model of feed-forward actor-critic (radial basis function) and ICO learning. However in this work we show that the combination of a recurrent neural network actor-critic with ICO learning, using the RMHP rule, was able to re-adapt the synaptic weights and combine the two systems effectively. The learned behavior greatly outperforms the

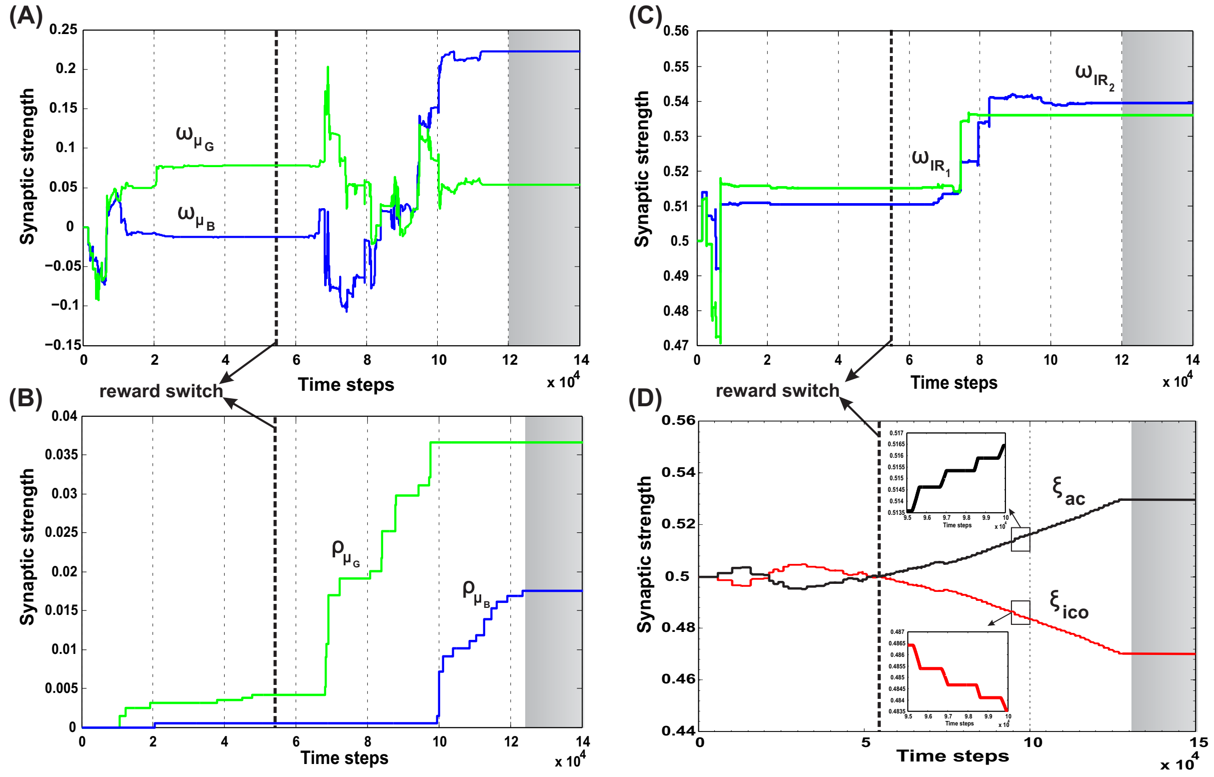


Figure 5.13: **Synaptic weight change curves for the dynamic foraging task** . **(A)** Change in the synaptic weights for actor-critic RL learner. Here w_{μ_G} corresponds to the input weights of the orientation sensor towards the green food source (spherical object) and w_{μ_B} corresponds to the input weights of the orientation sensor towards the blue. **(B)** Change in the synaptic weights of the ICO learner. ρ_{μ_G} - the CS stimulus weight for the orientation sensor towards green, ρ_{μ_B} the CS stimulus weight for the orientation sensor towards blue. **(C)** Change in the weights of the two infrared sensor inputs to the actor. w_{IR_1} - left IR sensor weight, w_{IR_2} - right IR sensor weights. Modulation of the IR sensor weights initially and during the periods 7×10^4 - 9×10^4 time steps can be attributed to the high degree of exploration during this time, where in the robot has considerable collisions with the boundary walls triggering these sensors (see Fig. 5.10 C). **(D)** Learning curve of the RMHP combined learning mechanism showing the change in the weights of the individual components. ξ_{ico} - weight of the ICO network output, ξ_{ac} - weight of the actor-critic RL network output. Here the ICO weights converge initially for the first part of the task, however fail to re-adapt upon change of reward signals. This is counter balanced by the correct evolution of the actor weights. As a result although initially the combinatorial learner places higher weight for the ICO network, after task switch, due to change in reinforcements the actor-critic RL system receives higher weights and drives the actual behavior of the robot. The inlaid plots show a magnified view of the two synaptic weights between 9.5×10^4 - 10×10^4. The plots show that the weights do not change in a fixed continuous manner, but increase/decrease in a step like formation corresponding to the specific points of reward activation (Fig. 5.12 A). In all the plots the grey shaded region mark the region of convergence for the respective synaptic weights, and the thick dashed line (black) marks the point of reward switch. (see text for more details).

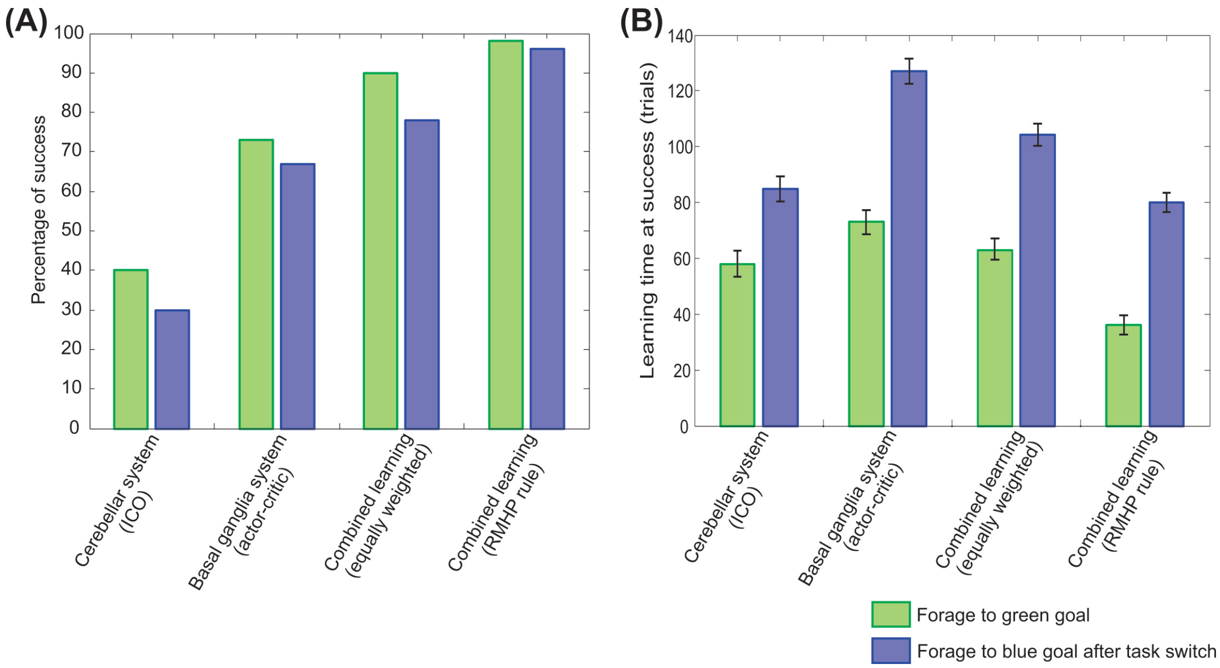


Figure 5.14: **Comparison of performance of RMHP modulated adaptive comninatorial learning system for the dynamic foraging task**. **(A)** Percentage of success measured over 50 experiments. **(B)** Average learning time (trials needed to successfully complete the task, calculated over 50 experiments (error bars indicate standard deviation with 98% confidence intervals). In both cases the green bars represent the performance for the initial task of learning to reach the green goal, while blue bars represent the performance in the subsequent task after dynamic switching of reward signals.

previous case and shows a high success rate for both, the initial navigation to green goal location and successively to the blue goal location, after switching of reinforcement signals.

In Fig. 5.14 B, we plot the average time taken to learn the first and second part of the dynamic foraging task. The learning time was calculated as the number of trials required on successful completion of the task (i.e successively reaching green or blue goal/food source location) averaged over 50 runs of the experiment. The combined learning mechanism with RMHP, successfully learns the task in less trials, as compared to the individual learning systems. However there was a significant increase in the learning time after the switching of reward signals. This can be attributed to the fact that after exploration goes to zero initially, a stable configuration is reached, the robot needs to perform more random explorations in order to change the strength of the synaptic connections considerably such that the opposite action of steering to the blue goal can be learned. Furthermore, as expected from the relatively fast learning rate of the ICO system, it was able to learn the tasks much quicker as compared to the actor-critic system, however its individual performance was less reliable than the actor-critic system as observed from the success rate (Fig. 5.14 A). Taken together, our model of RMHP induced combination mechanism provides a much more stable and fast decision making system as compared to the individual systems or a simple naive parallel combination of the two. At the same time the reservoir based actor-critic model also clearly outperforms (both in terms of stability and speed of learning), current state of the art feed-forward network models. Thus over all the neural combined learning mechanism with a self-adaptive reservoir critic mechanism, enables robust goal-directed learning with continuous time varying environmental stimuli.

5.4 Discussion

Numerous animal behavioral studies (Brembs and Heisenberg, 2000), (Lovibond, 1983), (Barnard, 2004) have pointed to an interactive role of classical and operant conditioning in guiding the decision making process for goal-directed learning. Typically a number of these psychology experiments reveal compelling evidence that both birds and mammals, can effectively learn to perform sophisticated tasks when trained using a combination of these mechanisms (Staddon, 1983), (Pierce and Cheney, 2013), (Shettleworth, 2009). The feeding behavior of Aplysia have also been used as model systems in order to compare classical and operant conditioning at the cellular level (Baxter and Byrne, 2006) (Brembs et al., 2004) and also study how predictive memory can be acquired by the neuronal correlates of the two learning paradigms (Brembs et al., 2002).
In case of the mamalian brain recent experimental evidence (Bostan et al., 2010), (Neychev et al., 2008) point towards the existence of direct communication and interactive combination between the neural substrates of reward learning and delay conditioning learning systems, namely the basal ganglia and the cerebellum. However the exact mechanism by which these two neural systems interact is still largely unknown. Few experimental studies suggest that such a communication could exists via the thalamus (Sakai et al., 2000), through which reciprocal connections from these two areas connect with the cortical areas in the brain (see Fig. 5.1) (Akkal et al., 2007), (McFarland and Haber, 2002). As such, in this paper we make the hypothesis (neural

combined learning) that such a combination is driven by a reward modulated heterosynaptic plasticity (Legenstein et al., 2008), (Hoerzer et al., 2012), triggered by dopaminergic projections (Varela, 2014), (García-Cabezas et al., 2007) existing at the thalamus that dynamically combines the output from the two areas and drives the overall goal directed behavior of an organism. It is important to note that, it is also possible that thalamic projections carrying basal-ganglia and cerebellar inputs could eventually converge onto a single pyramidal cell via relay neurons at the motor cortex. Furthermore, as the motor and frontal cortical regions together with the striatum, have been observed to receive particularly dense dopaminergic projections from the mid brain areas (VTA) (Hosp et al., 2011), it is plausible that the proposed neuromodulatory heterosynaptic plasticity could also occur directly at the cortex (Ni et al., 2014). We model the classical delay conditioning paradigm observed in the cerebellum with the help of input correlation learning (Porr and Wörgötter, 2006), while reward based learning modulated by prediction errors, is modeled using a temporal difference model of actor-critic learning. Using a simple robot model, and three different scenarios of increasing complexity for a foraging task, we demonstrate that the neural combinatorial learning mechanism can effectively and robustly enable the robot to move towards a desired food source while learning to avoid a negatively rewarded, undesired food source while being considerably robust to dynamic changes in the environmental setup.

Although there have been a few robot studies, trying to model basal ganglia behavior (Gurney et al., 2004), (Prescott et al., 2006) and cerebellar learning for classical conditioning (Verschure and Mintz, 2001), (Hofstoetter et al., 2002), to the best of our knowledge they have only been applied individually. In this study, for the first time, we show how such a combined mechanism can be implemented using a wheeled robot that leads to a more efficient decision making strategy. Although designed with a simplified level of biological abstraction, our model sheds light towards the way basal gangliar and cerebellar structures in the brain indirectly interact with each other through sensory feedback and partake in the processing of temporal environmental information in order to make decisions. Our model of the critic based on the self-adaptive reservoir network, takes into account evidence of both Hebbian plasticity and non-Hebbian homeostatic plasticity in the cortico-striatal synapses (Fino et al., 2005), as well as the strong reciprocal recurrent connections in the cortex that provide input to the striatal system (this is analogous to the output layer in our model) while being modulated by dopaminergic neural activity (TD-error). Static reservoir models of the basal ganglia system have been previously implemented in the context of learning language accusation (Hinaut and Dominey, 2013) or for modeling the experimentally observed timescales of neural activity of domapinergic neurons (Bernacchia et al., 2011). However, specifically in this work, the adaptive reservoir, not only provides a fading memory of incoming sensory stimuli that can enable the robot to deal with partially observable state space problems, but as demonstrated in the previous chapters (2 & 3), input dependent plastic changes in the network parameters allow optimal temporal information processing. As a result such a recurrently connected network clearly outperforms nonlinear feed-forward models of the critic (Morimoto and Doya, 1998). Furthermore, our work with the reservoir based critic sheds new insights in to how large recurrent networks can be trained in a non-supervised manner using reward modulation and a simple recursive least squares algorithm, which has hitherto been a difficult problem, with only few simple models existing that work on synthetic data (Hoerzer et al., 2012) or require supervised components (Koprinkova-Hristova et al., 2010).

In the context of goal directed behavior, one may also draw similarity of the basic reflexive mechanism learned by the cerebellum (Yeo and Hesslow, 1998) to innate or intrinsic motivations in biological organisms, in contrast to more extrinsic motivations (in the form of reinforcing evaluative feedbacks) provided by the striatal dopaminergic system of the basal ganglia (Boedecker et al., 2013). Our hypothesis is that in order for an organism to make decisions in a dynamic environment, where in, certain behaviors result in basic reflexes (based on CS - US conditioning) while others lead to specific rewards or punishments, it needs a mechanism that can effectively combine these, in order to accomplish the desired goal. Our neuromodulation scheme, namely, the RMHP rule provides such an adaptive combination that guides the behavior of the robot over time in order to achieve stable goal directed objectives. Particularly, our RMHP based combined learning model provides evidence that cooperation between reinforcement learning and correlation learning systems can enable agents to perform fast and stable reversal learning (adaptation to dynamic changes in the environment). Such combination mechanisms could be crucial in dealing with navigation scenarios involving contrasting or competing goals, with gradual or sudden changes to environmental conditions. Furthermore, this could also point towards possible adaptation or mal-adaptation between the basal ganglia and cerebellum in case of neurological movement disorders like dystonia (Neychev et al., 2008) which typically involve both these brain structures.

Over all our computational model based on the combinatorial learning hypothesis (Dasgupta et al., 2014b) shows that indeed the learning systems of the basal ganglia and the cerebellum can adaptively balance the output of each other in order to deal with changes in environment, reward conditions, and dynamic modulation of pre-learned decisions. Although here we modeled a novel reward modulation between the two systems, no direct feedback (interaction) between the cerebellum and basal ganglia was provided. In the future we plan to include such direct communication between the two in the form of inhibitory feedback, as evident from recent experimental studies (Bostan et al., 2010). In its current form, we envision such an adaptive combinatorial learning approach, coupled with the power of SARN to inherently encode time-varying information, to have wide impact on bio-mimetic agents, in order to provide better solutions of decision making problems in both static and dynamic situations, as well as show how the neuromodulation of executive circuits in the brain can effectively balance output from different areas. While our combined learning model verifies that the adaptive combination of the learning systems of the basal ganglia and the cerebellum leads to effective goal-directed behavior control in an artificial system, it would be interesting to further investigate this combination in biological systems, particularly in terms of the specific, underlying neuronal correlates.

CHAPTER 6

Discussion and Outlook

> ”I know why there are so many people who love chopping wood. In this activity one immediately sees the results”.
>
> —*Albert Einstein*

Each previous chapter contained its own extensive ’Discussion’ section (see sections 3.4, 4.5 and 5.4) where we compared our methods to other approaches and on occasion, related them to biological data. In this chapter we will, thus, only briefly summarize presented work by highlighting the main findings and possible limitations, provide directions for future work and conclude this thesis.

In this thesis we have focused on the topic of temporal information processing in the brain that also leads to memory guided behaviors, using a closed-loop system approach. In this regard, we showed that using an input-driven recurrent neural network model (as abstraction of abundant recurrent neural circuitry in the brain) with novel, local, self-adaptive or plastic processes, enables it to robustly perform such temporal processing. In the first part of the thesis, chapter 2, we introduced the concept of input driven recurrent networks (RNN) from the point of view of non-autonomous dynamical systems. We demonstrated that a generic class of such RNN models can be used to approximate to arbitrary levels of accuracy, any finite time trajectory of time-varying dynamical system. Considering the brain receives a constant barrage of complex time-varying stimuli and needs to compute with them, our model based on input-driven RNN forms an ideal setup to investigate the underlying processes for such temporal processing. Furthermore, our network model (also referred to as self-adaptive reservoir) forms a special case of this generic class of RNN models. In this chapter, we introduced two novel forms of local adaptations at the level of single neurons of the recurrent network. Firstly, an adaptation technique for the modulation of neuronal timeconstants or decay rates was presented, based on a new information theoretic measure called local active information storage. This allows each individual neurons to adapt their timescale with respect to the timescales of input signals and their own history of activity (local memory). Secondly, we derived a generalized intrinsic plasticity mechanism based on an optimal Weibull output distribution, in order, to tune the parameters that control the shape and scale of individual neuron non-linearity inside the network. As a result, the network was able to maintain homeostasis of neuronal activity, while at the same time allowing maximum information flow between input and output of each neuron. Finally, we combined this with a a supervised plasticity mechanism to adapt both the connection strengths inside the recurrent network as well as the connections from the recurrent network to readout neurons, which were trained for specific temporal tasks. Overall, this presents a novel self-adaptive reservoir model (SARN) for which we demonstrated, significantly superior performance as compared to static RNN models using an initial signal modeling task with inherently slow and fast timescales.

In chapter 3, we presented further elaborate experimental results demonstrating the superior performance of SARN in three different time-series benchmark tests, involving both non-linear computational power and temporal memory capacity. Furthermore, using delay embedded patterns generated by the Mackey-Glass time series, it was shown that, unlike non-adaptive static networks, SARN could learn to generate both stable as well as chaotic patterns. This occurs naturally from its intrinsic dynamics, in an input dependent manner. Post-adaptation, the over all network dynamics resides in a near critical region (edge of chaos), which leads to optimal processing of temporal information in the network. Using a clock-like, time interval processing task, we show that our network reproduces a linear increase in temporal variability with increase in square of the interval duration. This correlation that has been widely observed in experimental data (Ivry and Hazeltine, 1995), is well captured by our model, and as such, it shows that local plasticity or adaptation mechanisms may indeed be responsible for time perception in the brain (atleast, in the fast timescale of milliseconds to minutes). Of course, time perception in the brain, does not occur in isolation, but is intricately related to forms of behavior and memory. Therefore, using a closed-loop system with a complex walking robot, we demonstrated the application and superior performance of SARN on a delayed T-maze navigation task. Specifically, this required the maintenance of temporal stimuli (cue signals) and then later recall these at the T-junction, after varying delay periods, in order to make corresponding decisions. As such, this displayed SARN's ability to, not only generate precisely timed outputs, but also achieve extended memory of time-varying stimuli, that guides future behaviors. Finally, we also demonstrated that our local adaptation mechanisms complement the inherent transient dynamics of the network, by successfully learning a complex time dependent motor behavior like handwriting generation. Briefly active input stimuli were successful in enabling stable dynamic attractors (trajectories) in the high dimensional network state space, such that, motor patterns can be generated even in the presence of relatively high levels of perturbations. These results were compared with two state of the art RNN models, namely static chaotic RNN (Sussillo and Abbott, 2009) and a more recent 'innate trained' plastic RNN model (Laje and Buonomano, 2013), demonstrating SARN's superior performance in both cases. In essence, our results clearly indicate that homeostatic mechanisms like intrinsic plasticity and local neuronal timescale adaptations, can indeed enable robust learning of temporal patterns and memory guided behaviors, in a biologically plausible manner.

Motor prediction and planning is largely dependent on robust temporal information processing in the brain, especially in the milliseconds to seconds timescale. Furthermore due to the inherent delays in sensory information, internal forward models (Wolpert et al., 1998) are believed to be a mechanism by which the brain overcomes these delays and makes predictions of future motor signals. The modeling of such a predictive mechanism needs an intrinsic memory of recent motor commands, and an ability to adapt with changing sensory feedback signals. As such, in Chapter 4, we presented a novel neural mechanism to combine motor patterns generated by the central nervous system of a bio-inspired walking robot with internal forward models, based on our self-adaptive reservoir network. By designing SARN based forward models that works for each leg of a hexapod robot, in a distributed architecture, we clearly demonstrate the ability of our adaptive network to perform robust motor predictions. This enabled the walking robot to generate complex locomotive behaviors like climbing over large obstacles, crossing gaps and navigate uneven terrains. Furthermore, comparison with previous simple recurrent neuron forward models (Manoonpong et al., 2013b) demonstrate that SARN enables the robot to learn such predictive behaviors, in a much faster and stable manner. These results, highlight the robust performance of SARN, as well as, show a crucial link between the effect of plasticity in recurrent networks and the resulting adaptive behaviors. Moreover, unlike previous plastic RNN models (Toutounji and Pipa, 2014), (Lazar et al., 2009), which have been mainly applied to synthetic time-series modeling tasks; here, we demonstrate that the performance gained by SARN over static RNN models in synthetic tasks, can be easily transferred to complex engineering problems.

Finally, in chapter 5 we extend the previously presented supervised learning setup of SARN, to a more generic reward-based learning mechanism. This is crucial, since for biological systems, evaluative feedbacks from the environment in the form of rewards or punishments form an important component for developing conditioned responses. In this regard, here we specifically present a temporal-difference learning mechanism (Wörgötter and Porr, 2005) for adapting the synaptic connections from the recurrent layer in SARN to the readout neurons, in order to, learn some goal-directed behaviors. This is motivated as an abstract model of the basal-ganglia neural circuitry. Furthermore, in line with recent experimental evidences for the cooperative role of the striatal and cerebellar systems (Bostan et al., 2010), we show that, the reservoir reward learning system in combination with a correlation learning based model of the cerebellum, can lead to more efficient and stabler goal-directed decisions. We also introduced a novel biologically plausible, reward modulated heterosynaptic plasicity rule that can perform such a combined learning. Furthermore, it is clearly demonstrated that SARN outperforms traditional feed-forward neural network models for reward learning, specially in scenarios with inherent dependence on memory of incoming time-varying stimuli. Overall, the results obtained in this chapter clearly motivate the neurobiological grounding of our adaptive model from a realistic reward learning perspective, that can work in conjunction with other unsupervised learning strategies in the brain.

Outlook and Future Work

Understanding temporal information processing and dynamics of memory underlying large recurrent neural networks is essential, since this could explain the relationship between such dynamics or processes, and the resultant timing mechanisms in the brain. As such, several recent works have used artificial recurrent neural networks to model cortical functions like memory and learning that could underlie the brains ability to tell time (Buonomano and Maass, 2009), (Buonomano and Laje, 2010). The model presented in this thesis, also follows a similar approach, arguing that temporal processing or timing is inherently generated by the dynamics of large recurrently connected neurons (Dasgupta et al., 2014a). However, although some previous models have been able to capture the results obtained from time perception based psychophysical studies (Karmarkar and Buonomano, 2007), they have hitherto, not been applied for complex temporal processing tasks, especially in realistic closed-loop scenarios. In this thesis, we bridge this apparent gap of knowledge by introducing local adaptation and homeostatic plasticity in the recurrent network. As a result, we demonstrate for the first time, that post adaptation, initially random recurrent networks can be trained to generate complex temporal patterns and long memory of incoming stimuli. Different inputs can result in distinct locally stable or chaotic trajectories through the network space that is responsible for such robust temporal processing. These can then be used to generate complex memory guided behaviors in artificial agents. Previous work on similar self-organized adaptation of RNNs with a combination of homeostatic plasticity and synaptic plasticity, have either used simplistic binary neuron models with applications only on simple time-series data (Lazar et al., 2009), (Toutounji and Pipa, 2014), or optimized the network based on specific type of neuron non-linearity (radial-basis function units) (Lukoševicius, 2010). Furthermore, all such models have been formulated within a supervised learning paradigm. As such, these models fail to explain the information processing ability in the brain that lead to complex sensorimotor processing. In comparison, the SARN model presented in this thesis, shows that intrinsic plasticity and neuronal time constant modulation can not only adapt the network in a self-organized manner, but also generate complex behaviors in a biologically plausible way. This works not only in presence of specific teacher signals required for supervised learning, but also in the presence of realistic reinforcements (rewards or punishments) from the environment, based on instrumental conditioning in the brain. Moreover, the combination of IP, and local active information storage, based neuron time constant adaptation can be seen as a direct intervention at the algorithmic level of neural computation (Marr's levels of computation (Marr, 1982)). Such that, the level of information storage in the system can be adjusted in order to affect the higher-level computational goal of enhanced performance, on time processing tasks requiring large delay memory capacity.

However, it should be noted that an unrealistic aspect of our current model is the use of hyperbolic tangential neurons which can produce both positive and negative firing activity. Although this works best for computational purposes, from a biological perspective this violates Dale's principle, according to which neurons can be either excitatory or inhibitory, but not of a mixed type. This could be ameliorated by the use of more realistic sigmoidal neurons (activity $\in [0, 1]$) with separate excitatory and inhibitory types. As the intrinsic plasticity scheme presented in this thesis is based on a generic Weibull distribution, this could easily be adjusted by appropriately selecting the shape (α) and scale (β) parameters of the distribution, in order to account for the sigmoidal non-linearity shape.

Although in our present model, local adaptations and plasticity mechanisms provide guided self-organization (Prokopenko, 2009), leading to significant improvement to the temporal information processing capability of RNNs, a number of open questions remains that can be addressed with future work. Some of them being:

- The reservoir models presented in here, demonstrate that complex (transient) dynamics enables neural circuits to process a broad range of nonlinear, temporal problems. This is largely based on the inherent stochasticity (randomness) of neural systems and local adaptation processes that occur at a fast timescale of milliseconds to seconds (like the ones presented in this thesis). However, it is known that slow synaptic plasticity mechanisms (minutes to hours) in the brain adapt synaptic efficacies to form ordered structures - called cell assemblies. These form strongly interconnected clusters of neurons that can encode associative memories of previously experiences stimuli. Typically this type of dynamics is characterized by persistent or steady state activity, and is in stark contrast to the transient dynamics of randomly connected reservoir networks. Intriguingly, several experiments show that both concepts coexist in the same neural circuits (Fujisawa et al., 2008), (Mante et al., 2013). As such, this begs the question, how this coexistence of transient dynamics and cell assemblies can emerge in one neural circuit? (Tetzlaff et al., 2014) and what role does local adaptations and homeostatic plasticity play in this process? Extension of the current model using a combination of Hebbian synaptic plasticity (instead of the supervised learning presented here) along with the homeostatic plasticity mechanisms, would be needed to address these questions.

- Here we extended the SARN model to work from within a reward learning paradigm. Typically the synaptic connections from the reservoir or recurrent layer was learned using a neuromodulatory signal based on the temporal difference error. Such a neuromodulatory, reward prediction error signal are believed to be encoded primarily by dopaminergic neurons (Schultz and Dickinson, 2000). Experimental evidence suggests that dopaminergic neuromodulatory signals, typically modulate synaptic plasticity in the brain, specifically in the prefrontal cortex (Otani et al., 2003). Therefore, it is plausible that such modulation of synaptic efficacies occurs within the recurrent layer of our model, so as to bring about task specific representations. In future work, the interaction of such modulatory processes with fast homeostatic and slow synaptic plasticity mechanisms within the dynamic reservoir could be investigated.

- Finally, although here we specifically focused on the active information storage at single neurons, in order to adapt their decay rates or time constants; neural activity in networks is not only dependent on its own previous history but also the flow of information from neighboring neurons. The use of measures like transfer entropy or Granger causality help quantify such information. Therefore, it would be interesting to use both transfer and storage measures to adapt neuronal decay rates. Such a combination could provide a measure for the true modified information at a single neuron level as a result of changes in the timescale of inputs.

Overall, the work presented in this thesis provides the crucial link between homeostatic mechanisms and local unsupervised adaptation processes in neuronal networks and their effect on the networks ability to perform complex temporal information processing. This also forms a novel self-adaptive framework for modeling time perception and related behaviors in the brain. Furthermore, using a closed-loop approach, it clearly demonstrates how robust memory guided behaviors can be generated from the resultant transient dynamics of such an adaptive network.

This was hitherto not shown in static recurrent neural network models. As such, it lays the foundation for future work, that can answer some of the critical questions raised above.

BIBLIOGRAPHY

Abbott, L. F. and Nelson, S. B. (2000). Synaptic plasticity: Taming the beast. *Nat. Neurosci. (Suppl.)*, 3:1178–1183.

Akkal, D., Dum, R. P., and Strick, P. L. (2007). Supplementary motor area and presupplementary motor area: targets of basal ganglia and cerebellar output. *The Journal of Neuroscience*, 27(40):10659–10673.

Allen, G. and Tsukahara, N. (1974). Cerebrocerebellar communication systems. *Physiological Reviews*, 54(4):957–1006.

Anderson, M. E. and Turner, R. S. (1991). Activity of neurons in cerebellar-receiving and pallidal-receiving areas of the thalamus of the behaving monkey. *Journal of Neurophysiology*, 66(3):879–893.

Antonelo, E., Schrauwen, B., and Stroobandt, D. (2008). Mobile robot control in the road sign problem using reservoir computing networks. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 911–916. IEEE.

Arkin, R. C. (1998). *Behavior-based robotics.* MIT press.

Bailey, C. H., Giustetto, M., Huang, Y.-Y., Hawkins, R. D., and Kandel, E. R. (2000). Is heterosynaptic modulation essential for stabilizing hebbian plasiticity and memory. *Nature Reviews Neuroscience*, 1(1):11–20.

Barnard, C. J. (2004). *Animal behaviour: mechanism, development, function and evolution.* Pearson Education.

Baxter, D. A. and Byrne, J. H. (2006). Feeding behavior of aplysia: a model system for comparing cellular mechanisms of classical and operant conditioning. *Learning & Memory*, 13(6):669–680.

Beer, R. D. and Ritzmann, R. E. (1993). *Biological neural networks in invertebrate neuroethology and robotics.* Academic Pr.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *Neural Networks, IEEE Transactions on*, 5(2):157–166.

Bernacchia, A., Seo, H., Lee, D., and Wang, X.-J. (2011). A reservoir of time constants for memory traces in cortical neurons. *Nature neuroscience*, 14(3):366–372.

Bi, G. Q. and Poo, M. M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.*, 18(24):10464–10472.

Bialek, W., Nemenman, I., and Tishby, N. (2001). Complexity through nonextensivity. *Physica A: Statistical Mechanics and its Applications*, 302(1):89–99.

Blaesing, B. and Cruse, H. (2004). Stick insect locomotion in a complex environment: climbing over large gaps. *Journal of experimental biology*, 207(8):1273–1286.

Bliss, T. and Lomo, T. (1973). Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. *J. Physiol.*, 232:331–356.

Boedecker, J., Lampe, T., and Riedmiller, M. (2013). Modeling effects of intrinsic and extrinsic rewards on the competition between striatal learning systems. *Frontiers in psychology*, 4.

Boedecker, J., Obst, O., Lizier, J. T., Mayer, N. M., and Asada, M. (2012). Information processing in echo state networks at the edge of chaos. *Theory in Biosciences*, 131(3):205–213.

Boedecker, J., Obst, O., Mayer, N. M., and Asada, M. (2009). Initialization and self-organized optimization of recurrent neural network connectivity. *HFSP journal*, 3(5):340–349.

Bosch-Bouju, C., Hyland, B. I., and Parr-Brownlie, L. C. (2013). Motor thalamus integration of cortical, cerebellar and basal ganglia information: implications for normal and parkinsonian conditions. *Frontiers in computational neuroscience*, 7.

Bostan, A. C., Dum, R. P., and Strick, P. L. (2010). The basal ganglia communicate with the cerebellum. *Proceedings of the National Academy of Sciences*, 107(18):8452–8456.

Boyd, S. and Chua, L. O. (1985). Fading memory and the problem of approximating nonlinear operators with volterra series. *Circuits and Systems, IEEE Transactions on*, 32(11):1150–1161.

Braun, J. M., Wörgötter, F., and Manoonpong, P. (2014). Internal models support specific gaits in orthotic devices. In *Mobile Service Robotics*, number 17 in Proceedings of the International Conference on Climbing and Walking Robots, pages 539–546.

Brembs, B., Baxter, D. A., and Byrne, J. H. (2004). Extending in vitro conditioning in aplysia to analyze operant and classical processes in the same preparation. *Learning & memory*, 11(4):412–420.

Brembs, B. and Heisenberg, M. (2000). The operant and the classical in conditioned orientation of drosophila melanogaster at the flight simulator. *Learning & Memory*, 7(2):104–115.

Brembs, B., Lorenzetti, F. D., Reyes, F. D., Baxter, D. A., and Byrne, J. H. (2002). Operant reward learning in aplysia: neuronal correlates and mechanisms. *Science*, 296(5573):1706–1709.

Bressler, S. L. and Kelso, J. (2001). Cortical coordination dynamics and cognition. *Trends in cognitive sciences*, 5(1):26–36.

Brons, J. F. and Woody, C. D. (1980). Long-term changes in excitability of cortical neurons after pavlovian conditioning and extinction. *J. Neurophysiol*, 44(3):605–615.

Bueti, D. and Walsh, V. (2009). The parietal cortex and the representation of time, space, number and other magnitudes. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1525):1831–1840.

Buhusi, C. V. and Meck, W. H. (2005). What makes us tick? functional and neural mechanisms of interval timing. *Nature Reviews Neuroscience*, 6:755–765.

Buonomano, D. V. (2000). Decoding temporal information: a model based on short-term synaptic plasticity. *The Journal of Neuroscience*, 20(3):1129–1141.

Buonomano, D. V. (2007). The biology of time across different scales. *Nature chemical biology*, 3(10):594–597.

Buonomano, D. V., Bramen, J., and Khodadadifar, M. (2009). Influence of the interstimulus interval on temporal processing and learning: testing the state-dependent network model. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364:1865–1873.

Buonomano, D. V. and Laje, R. (2010). Population clocks: motor timing with neural dynamics. *Trends in cognitive sciences*, 14(12):520–527.

Buonomano, D. V. and Maass, W. (2009). State-dependent computations: spatiotemporal processing in cortical networks. *Nat. Rev. Neurosci.*, 10:113–125.

Burguiere, E., Arabo, A., Jarlier, F., Zeeuw, C. I. D., and Rondi-Reig, L. (2010). Role of the cerebellar cortex in conditioned goal-directed behavior. *The Journal of Neuroscience*, 30(40):13265–13271.

Burrone, J., O'Byrne, M., and Murthy, V. N. (2002). Multiple forms of synaptic plasticity triggered by selective suppression of activity in individual neurons. *Nature*, 420:414–418.

Bush, K. and Anderson, C. (2005). Modeling reward functions for incomplete state representations via echo state networks. In *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, volume 5, pages 2995–3000. IEEE.

Buteneers, P., Schrauwen, B., Verstraeten, D., and Stroobandt, D. (2009). Real-time epileptic seizure detection on intra-cranial rat data using reservoir computing. In *Advances in neuro-information processing*, pages 56–63. Springer.

Carla Shatz, J. (1992). The developing brain. *Sci. Am.*, 267:60–67.

Chistiakova, M. and Volgushev, M. (2009). Heterosynaptic plasticity in the neocortex. *Experimental brain research*, 199(3-4):377–390.

Chow, T. W. and Li, X.-D. (2000). Modeling of continuous time dynamical systems with input by recurrent neural networks. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 47(4):575–578.

Christian, K. M. and Thompson, R. F. (2003). Neural substrates of eyeblink conditioning: acquisition and retention. *Learning & memory*, 10(6):427–455.

Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Ryu, S. I., and Shenoy, K. V. (2010). Cortical preparatory activity: representation of movement or first cog in a dynamical machine? *Neuron*, 68(3):387–400.

Citri, A. and Malenka, R. C. (2007). Synaptic plasticity: multiple forms, functions, and mechanisms. *Neuropsychopharmacology*, 33(1):18–41.

Clark, R. E. and Squire, L. R. (1998). Classical conditioning and brain systems: The role of awareness. *Science*, 280(5360):77–81.

Cohen, J. Y., Haesler, S., Vong, L., Lowell, B. B., and Uchida, N. (2012). Neuron-type-specific signals for reward and punishment in the ventral tegmental area. *Nature*, 482(7383):85–88.

Cruse, H. (1976). The control of body position in the stick insect (carausius morosus), when walking over uneven surfaces. *Biological Cybernetics*, 24(1):25–33.

Crutchfield, J. P. and Feldman, D. P. (2003). Regularities unseen, randomness observed: Levels of entropy convergence. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 13(1):25–54.

Crutchfield, J. P. and Young, K. (1989). Inferring statistical complexity. *Physical Review Letters*, 63(2):105.

Daoudal, G. and Debanne, D. (2003). Long-term plasticity of intrinsic excitability: learning rules and mechanisms. *Learning & Memory*, 10(6):456–465.

Dasgupta, S., Manoonpong, P., and Wörgötter, F. (2014a). Reservoir of neurons with adaptive time constants: a hybrid model for robust motor-sensory temporal processing. *BMC Neuroscience*, 15(Suppl 1):P9.

Dasgupta, S., Wörgötter, F., and Manoonpong, P. (2012). Information theoretic self-organised adaptation in reservoirs for temporal memory tasks. In *Engineering Applications of Neural Networks*, pages 31–40. Springer.

Dasgupta, S., Wörgötter, F., and Manoonpong, P. (2013a). Information dynamics based self-adaptive reservoir for delay temporal memory tasks. *Evolving Systems*, 4(4):235–249.

Dasgupta, S., Wörgötter, F., and Manoonpong, P. (2014b). Neuromodulatory adaptive combination of correlation-based learning in cerebellum and reward-based learning in basal ganglia for goal-directed behavior control. *Frontiers in neural circuits*, 8.

Dasgupta, S., Wörgötter, F., Morimoto, J., and Manoonpong, P. (2013b). Neural combinatorial learning of goal-directed behavior with reservoir critic and reward modulated hebbian plasticity. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pages 993–1000.

Davis, G. W. (2006). Homeostatic control of neural activity: from phenomenology to molecular design. *Annu. Rev. Neurosci.*, 29:307–323.

Dayan, P. and Abbott, L. (2003). Theoretical neuroscience: computational and mathematical modeling of neural systems. *Journal of Cognitive Neuroscience*, 15(1):154–155.

Dayan, P. and Balleine, B. W. (2002). Reward, motivation, and reinforcement learning. *Neuron*, 36(2):285–298.

Dearden, A. and Demiris, Y. (2005). Learning forward models for robots. In *International Joint Conference on Artificial Intelligence*, volume 5, page 1440.

Desai, N. S., Rutherford, L. C., and Turrigiano, G. G. (1999). Plasticity in the intrinsic excitability of cortical pyramidal neurons. *Nat. Neurosci.*, 2(6):515–520.

Desiraju, T. and Purpura, D. (1969). Synaptic convergence of cerebellar and lenticular projections to thalamus. *Brain Research*, 15(2):544–547.

Douglas, R. J. and Martin, K. A. (2004). Neuronal circuits of the neocortex. *Annu. Rev. Neurosci.*, 27:419–451.

Doya, K. (1999). What are the computations of the cerebellum, the basal ganglia and the cerebral cortex? *Neural networks*, 12(7):961–974.

Doya, K. (2000a). Complementary roles of basal ganglia and cerebellum in learning and motor control. *Current opinion in neurobiology*, 10(6):732–739.

Doya, K. (2000b). Reinforcement learning in continuous time and space. *Neural computation*, 12(1):219–245.

Dreher, J.-C. and Grafman, J. (2002). The roles of the cerebellum and basal ganglia in timing and error prediction. *European Journal of Neuroscience*, 16(8):1609–1619.

Dudai, Y. (2004). The neurobiology of consolidation, or, how stable is the engram? *Annu. Rev. Psychol.*, 55:51–86.

Dudek, S. M. and Bear, M. F. (1992). Homosynaptic long-term depression in area CA1 of hippocampus and effects of N-methyl-D-aspartate receptor blockade. *Proc. Natl. Acad. Sci. USA*, 89:4363–4367.

Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2):179–211.

Elman, J. L. and Zipser, D. (1988). Learning the hidden structure of speech. *The Journal of the Acoustical Society of America*, 83(4):1615–1626.

Fino, E., Glowinski, J., and Venance, L. (2005). Bidirectional activity-dependent plasticity at corticostriatal synapses. *The Journal of neuroscience*, 25(49):11279–11287.

Freeman, J. H. and Steinmetz, A. B. (2011). Neural circuitry and plasticity mechanisms underlying delay eyeblink conditioning. *Learning & Memory*, 18(10):666–677.

Fremaux, N., Sprekeler, H., and Gerstner, W. (2013). Reinforcement learning using a continuous time actor-critic framework with spiking neurons. *PLoS computational biology*, 9(4).

Fujisawa, S., Amarasingham, A., Harrison, M. T., and Buzsáki, G. (2008). Behavior-dependent short-term assembly dynamics in the medial prefrontal cortex. *Nature neuroscience*, 11(7):823–833.

Funahashi, K.-i. and Nakamura, Y. (1993). Approximation of dynamical systems by continuous time recurrent neural networks. *Neural networks*, 6(6):801–806.

Gandhi, C. C. and Matzel, L. D. (2000). Modulation of presynaptic action potential kinetics underlies synaptic facilitation of type b photoreceptors after associative conditioning in hermissenda. *The Journal of Neuroscience*, 20(5):2022–2035.

Ganguli, S., Huh, D., and Sompolinsky, H. (2008). Memory traces in dynamical systems. *Proceedings of the National Academy of Sciences*, 105(48):18970–18975.

Ganguly, K. and Carmena, J. M. (2009). Emergence of a stable cortical map for neuroprosthetic control. *PLoS biology*, 7(7):e1000153.

García-Cabezas, M. Á., Rico, B., Sánchez-González, M. Á., and Cavada, C. (2007). Distribution of the dopamine innervation in the macaque and human thalamus. *Neuroimage*, 34(3):965–984.

Gerstner, W., Kempter, R., van Hemmen, J. L., and Wagner, H. (1996). A neuronal learning rule for sub-millisecond temporal coding. *Nature*, 383:76–78.

Glass, L. and Mackey, M. C. (1988). *From clocks to chaos: the rhythms of life*. Princeton University Press.

Goldman-Rakic, P. (1995). Cellular basis of working memory. *Neuron*, 14(3):477–485.

Goldschmidt, D., Wörgötter, F., and Manoonpong, P. (2014). Biologically-inspired adaptive obstacle negotiation behavior of hexapod robots. *Frontiers in neurorobotics*, 8.

Golub, M. D., Yu, B., and Chase, S. M. (2012). Internal models engaged by brain-computer interface control. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 1327–1330. IEEE.

Gouvea, T. S., Monteiro, T., Soares, S., Atallah, B. V., and Paton, J. J. (2014). Ongoing behavior predicts perceptual report of interval duration. *Frontiers in neurorobotics*, 8.

Gurney, K., Prescott, T. J., and Redgrave, P. (2001). A computational model of action selection in the basal ganglia. i. a new functional anatomy. *Biological cybernetics*, 84(6):401–410.

Gurney, K., Prescott, T. J., Wickens, J. R., and Redgrave, P. (2004). Computational models of the basal ganglia: from robots to membranes. *Trends in neurosciences*, 27(8):453–459.

Haber, S. N. and Calzavara, R. (2009). The cortico-basal ganglia integrative network: the role of the thalamus. *Brain research bulletin*, 78(2):69–74.

Hartland, C. and Bredeche, N. (2007). Using echo state networks for robot navigation behavior acquisition. In *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE International Conference on*, pages 201–206. IEEE.

Hazeltine, E., Helmuth, L. L., and Ivry, R. B. (1997). Neural mechanisms of timing. *Trends in Cognitive Sciences*, 1(5):163–169.

Hebb, D. O. (1949). *The Organization of Behaviour*. Wiley, New York.

Held, R. (1961). Exposure-history as a factor in maintaining stability of perception and coordination. *The Journal of nervous and mental disease*, 132(1):26–hyhen.

Hennequin, G., Vogels, T. P., and Gerstner, W. (2014). Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron*, 82(6):1394–1406.

Herreros, I. and Verschure, P. F. (2013). Nucleo-olivary inhibition balances the interaction between the reactive and adaptive layers in motor control. *Neural Networks*, 47:64–71.

Hinaut, X. and Dominey, P. F. (2013). Real-time parallel processing of grammatical structure in the fronto-striatal system: a recurrent network simulation study using reservoir computing. *PloS one*, 8(2).

Hinton, S. C. and Meck, W. H. (2004). Frontal–striatal circuitry activated by human peak-interval timing in the supra-seconds range. *Cognitive Brain Research*, 21(2):171–182.

Hoerzer, G. M., Legenstein, R., and Maass, W. (2012). Emergence of complex computational structures from chaotic neural networks through reward-modulated hebbian learning. *Cerebral Cortex*, 24(3):677–690.

Hofstoetter, C., Mintz, M., and Verschure, P. F. (2002). The cerebellum in action: a simulation and robotics study. *European Journal of Neuroscience*, 16(7):1361–1376.

Holst, E. and Mittelstaedt, H. (1950). Das reafferenzprinzip. *Naturwissenschaften*, 37(20):464–476.

Holtmaat, A. and Svoboda, K. (2009). Experience-dependent structural synaptic plasticity in the mammalian brain. *Nature Reviews Neuroscience*, 10(9):647–658.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA*, 79:2554–2558.

Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. USA*, 81:3088–3092.

Hoshi, E., Tremblay, L., Féger, J., Carras, P. L., and Strick, P. L. (2005). The cerebellum communicates with the basal ganglia. *Nature neuroscience*, 8(11):1491–1493.

Hosp, J. A., Pekanovic, A., Rioult-Pedotti, M. S., and Luft, A. R. (2011). Dopaminergic projections from midbrain to primary motor cortex mediate motor skill learning. *The Journal of Neuroscience*, 31(7):2481–2487.

Houk, J., Bastianen, C., Fansler, D., Fishbach, A., Fraser, D., Reber, P., Roy, S., and Simo, L. (2007). Action selection and refinement in subcortical loops through basal ganglia and cerebellum. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1485):1573–1583.

Houk, J. C., Adams, J. L., and Barto, A. G. (1995). A model of how the basal ganglia generate and use neural signals that predict reinforcement. In Houk, J. C., Davis, J. L., and Beiser, D. G., editors, *Models of information processing in the basal ganglia*, pages 249–270.

Huebner, U., Abraham, N., and Weiss, C. (1989). Dimensions and entropies of chaotic intensity pulsations in a single-mode far-infrared nh 3 laser. *Physical Review A*, 40(11):6354.

Humphries, M. D., Stewart, R. D., and Gurney, K. N. (2006). A physiologically plausible model of action selection and oscillatory activity in the basal ganglia. *The Journal of neuroscience*, 26(50):12921–12942.

Huston, S. J. and Jayaraman, V. (2011). Studying sensorimotor integration in insects. *Current opinion in neurobiology*, 21(4):527–534.

Ishikawa, M., Otaka, M., Huang, Y. H., Neumann, P. A., Winters, B. D., Grace, A. A., Schlüter, O. M., and Dong, Y. (2013). Dopamine triggers heterosynaptic plasticity. *The Journal of Neuroscience*, 33(16):6759–6765.

Ivry, R. B. and Hazeltine, R. E. (1995). Perception and production of temporal intervals across a range of durations: evidence for a common timing mechanism. *Journal of Experimental Psychology: Human Perception and Performance*, 21(1):3.

Ivry, R. B. and Spencer, R. (2004). The neural representation of time. *Current opinion in neurobiology*, 14:225–232.

Jaeger, H. (2001a). The echo state approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, 148:34.

Jaeger, H. (2001b). *Short term memory in echo state networks.* GMD-Forschungszentrum Informationstechnik.

Jaeger, H. (2014). Controlling recurrent neural networks by conceptors. *arXiv preprint arXiv:1403.3369*.

Jaeger, H. and Haas, H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667):78–80.

Jaeger, H., Lukoševičius, M., Popovici, D., and Siewert, U. (2007). Optimization and applications of echo state networks with leaky-integrator neurons. *Neural Networks*, 20(3):335–352.

Jarvis, S., Rotter, S., and Egert, U. (2010). Extending stability through hierarchical clusters in echo state networks. *Frontiers in neuroinformatics*, 4.

Jiang, F., Berry, H., and Schoenauer, M. (2008). Supervised and evolutionary learning of echo state networks. In *Parallel Problem Solving from Nature–PPSN X*, pages 215–224. Springer.

Jin, L., Nikiforuk, P. N., and Gupta, M. M. (1995). Approximation of discrete-time state-space trajectories using dynamic recurrent neural networks. *Automatic Control, IEEE Transactions on*, 40(7):1266–1270.

Joel, D., Niv, Y., and Ruppin, E. (2002). Actor-critic models of the basal ganglia: New anatomical and computational perspectives. *Neural networks*, 15(4):535–547.

Joel, D. and Weiner, I. (2000). The connections of the dopaminergic system with the striatum in rats and primates: an analysis with respect to the functional and compartmental organization of the striatum. *Neuroscience*, 96(3):451–474.

Jones, E. G., Steriade, M., and McCormick, D. (1985). *The thalamus*. Plenum Press New York.

Jordan, M. I. and Rumelhart, D. E. (1992). Forward models: Supervised learning with a distal teacher. *Cognitive science*, 16(3):307–354.

Kantz, H. (1994). A robust method to estimate the maximal lyapunov exponent of a time series. *Physics letters A*, 185(1):77–87.

Karmarkar, U. R. and Buonomano, D. V. (2007). Timing in the absence of clocks: encoding time in neural network states. *Neuron*, 53(3):427–438.

Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current opinion in neurobiology*, 9(6):718–727.

Kawato, M. (2008). From âĂŸunderstanding the brain by creating the brainâĂŹtowards manipulative neuroscience. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 363(1500):2201–2214.

Kawato, M., Kuroda, S., and Schweighofer, N. (2011). Cerebellar supervised learning revisited: biophysical modeling and degrees-of-freedom control. *Current opinion in neurobiology*, 21(5):791–800.

Kesper, P., Grinke, E., Hesse, F., Wörgötter, F., and Manoonpong, P. (2013). Obstacle/gap detection and terrain classification of walking robots based on a 2d laser range finder. *Chapter*, 53:419–426.

Kilman, V., van Rossum, M. C., and Turrigiano, G. G. (2002). Activity deprivation reduces miniature IPSC amplitude by decreasing the number of postsynaptic GABA_A receptors clustered at neocortical synapses. *J. Neurosci.*, 22:1328–1337.

Kim, J. J. and Thompson, R. E. (1997). Cerebellar circuits and synaptic mechanisms involved in classical eyeblink conditioning. *Trends in neurosciences*, 20(4):177–181.

Kitazawa, S., Kimura, T., and Yin, P.-B. (1998). Cerebellar complex spikes encode both destinations and errors in arm movements. *Nature*, 392(6675):494–497.

Kloeden, P. E. and Rasmussen, M. (2011). *Nonautonomous dynamical systems*, volume 176. American Mathematical Soc.

Knudsen, E. (1994). Supervised learning in the brain. *Journal of Neuroscience*, 14(7):3985–3997.

Koch, C., Rapp, M., and Segev, I. (1996). A brief history of time (constants). *Cerebral cortex*, 6(2):93–101.

Kolodziejski, C., Porr, B., and Wörgötter, F. (2008). Mathematical properties of neuronal td-rules and differential hebbian learning: a comparison. *Biological cybernetics*, 98(3):259–272.

Koprinkova-Hristova, P., Oubbati, M., and Palm, G. (2010). Adaptive critic design with echo state network. In *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*, pages 1010–1015.

Kraus, B. J., Robinson II, R. J., White, J. A., Eichenbaum, H., and Hasselmo, M. E. (2013). Hippocampal âĂIJtime cellsâĂİ: time versus path integration. *Neuron*, 78(6):1090–1101.

Kreitzer, A. C. and Malenka, R. C. (2008). Striatal plasticity and basal ganglia circuit function. *Neuron*, 60(4):543–554.

Krupa, D. J., Thompson, J. K., and Thompson, R. F. (1993). Localization of a memory trace in the mammalian brain. *Science*, 260(5110):989–991.

Kuramoto, E., Furuta, T., Nakamura, K. C., Unzai, T., Hioki, H., and Kaneko, T. (2009). Two types of thalamocortical projections from the motor thalamic nuclei of the rat: a single neuron-tracing study using viral vectors. *Cerebral Cortex*, 19(9):2065–2077.

Kuwabara, J., Nakajima, K., Kang, R., Branson, D. T., Guglielmino, E., Caldwell, D. G., and Pfeifer, R. (2012). Timing-based control via echo state network for soft robotic arm. In *Neural Networks (IJCNN), The 2012 International Joint Conference on*, pages 1–8. IEEE.

Laje, R. and Buonomano, D. V. (2013). Robust timing and motor patterns by taming chaos in recurrent neural networks. *Nature neuroscience*, 16(7):925–933.

Lau, C. G. and Zukin, R. S. (2007). Nmda receptor trafficking in synaptic plasticity and neuropsychiatric disorders. *Nature Reviews Neuroscience*, 8(6):413–426.

Lazar, A., Pipa, G., and Triesch, J. (2007). Fading memory and time series prediction in recurrent networks with different forms of plasticity. *Neural Networks*, 20:312–322.

Lazar, A., Pipa, G., and Triesch, J. (2009). Sorn: a self-organizing recurrent neural network. *Frontiers in computational neuroscience*, 3.

Lee, J.-W. and Lee, G.-K. (2005). Gait angle prediction for lower limb orthotics and prostheses using an emg signal and neural networks. *International Journal of Control, Automation, and Systems*, 3(2):152–158.

Legenstein, R., Chase, S. M., Schwartz, A. B., and Maass, W. (2010). A reward-modulated hebbian learning rule can explain experimentally observed network reorganization in a brain control task. *The Journal of Neuroscience*, 30(25):8400–8410.

Legenstein, R. and Maass, W. (2007a). Edge of chaos and prediction of computational performance for neural circuit models. *Neural Networks*, 20(3):323–334.

Legenstein, R. and Maass, W. (2007b). What makes a dynamical system computationally powerful. *New directions in statistical signal processing: From systems to brain*, pages 127–154.

Legenstein, R., Pecevski, D., and Maass, W. (2008). A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. *PLoS Computational Biology*, 4(10).

Levy, W. B. and Steward, O. (1983). Temporal contiguity requirements for long-term associative potentiation/depression in the hippocampus. *Neuroscience*, 8(4):791–797.

Lewis, P. and Miall, R. (2006). A right hemispheric prefrontal system for cognitive time measurement. *Behavioural Processes*, 71(2):226–234.

Lisberger, S. and Thach, T. (2013). The cerebellum. In Kandel, E. R., Schwartz, J. H., Jessell, T. M., et al., editors, *Principles of neural science*, pages 960–981. McGraw-Hill, New York.

Lizier, J. T. (2014). Jidt: An information-theoretic toolkit for studying the dynamics of complex systems. *Frontiers in Robotics and AI*, 1(11).

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2012). Local measures of information storage in complex distributed computation. *Information Sciences*, 208:39–54.

Loewenstein, Y. and Sompolinsky, H. (2003). Temporal integration by calcium dynamics in a model neuron. *Nature neuroscience*, 6(9):961–967.

Lonini, L., Dipietro, L., Zollo, L., Guglielmelli, E., and Krebs, H. I. (2009). An internal model for acquisition and retention of motor learning during arm reaching. *Neural computation*, 21(7):2009–2027.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the atmospheric sciences*, 20(2):130–141.

Lovibond, P. F. (1983). Facilitation of instrumental behavior by a pavlovian appetitive conditioned stimulus. *Journal of Experimental Psychology: Animal Behavior Processes*, 9(3):225–247.

Lukoševicius, M. (2010). On self-organizing reservoirs and their hierarchies. *Jacobs University Bremen, Tech. Rep*, (25).

Lukoševičius, M. and Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149.

Lynch, G. S., Dunwiddie, T., and Gribkoff, V. (1977). Heterosynaptic depression: a postsynaptic correlate of long-term potentiation. *Nature*, 266:737–739.

Maass, W., Natschlaeger, T., and Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, 14(11):2531–2560.

Maass, W., Natschläger, T., and Markram, H. (2004). Computational models for generic cortical microcircuits. *Computational neuroscience: A comprehensive approach*, pages 575–605.

Mackey, M. C., Glass, L., et al. (1977). Oscillation and chaos in physiological control systems. *Science*, 197(4300):287–289.

Magee, J. C. and Johnston, D. (1997). A synaptically controlled, associative signal for Hebbian plasticity in hippocampal neurons. *Science*, 275:209–213.

Manjunath, G. and Jaeger, H. (2013). Echo state property linked to an input: Exploring a fundamental characteristic of recurrent neural networks. *Neural computation*, 25(3):671–696.

Manoonpong, P., Dasgupta, S., Goldschmidt, D., and Worgotter, F. (2014). Reservoir-based online adaptive forward models with neural control for complex locomotion in a hexapod robot. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, pages 3295–3302. IEEE.

Manoonpong, P., Geng, T., Kulvicius, T., Porr, B., and Wörgötter, F. (2007). Adaptive, fast walking in a biped robot under neuronal control and learning. *PLoS Computational Biology*, 3(7):e134.

Manoonpong, P., Kolodziejski, C., Wörgötter, F., and Morimoto, J. (2013a). Combining correlation-based and reward-based learning in neural control for policy improvement. *Advances in Complex Systems*, 16(02n03).

Manoonpong, P., Parlitz, U., and Wörgötter, F. (2013b). Neural control and adaptive neural forward models for insect-like, energy-efficient, and adaptable locomotion of walking machines. *Frontiers in neural circuits*, 7.

Mante, V., Sussillo, D., Shenoy, K. V., and Newsome, W. T. (2013). Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84.

Markram, H., Lübke, J., Frotscher, M., and Sakmann, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, 275:213–215.

Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information, henry holt and co. *Inc., New York, NY*, pages 2–46.

Martin, S. J., Grimwood, P. D., and Morris, R. G. M. (2000). Synaptic plasticity and memory: an evaluation of the hypothesis. *Annu. Rev. Neurosci.*, 23:649–711.

Martin, S. J. and Morris, R. G. M. (2002). New life in an old idea: the synaptic plasticity and memory hypothesis revisited. *Hippocampus*, 12:609–636.

Mauk, M. D. and Buonomano, D. V. (2004). The neural basis of temporal processing. *Annu. Rev. Neurosci.*, 27:307–340.

McFarland, N. R. and Haber, S. N. (2002). Thalamic relay nuclei of the basal ganglia form both reciprocal and nonreciprocal cortical connections, linking multiple frontal cortical areas. *The Journal of neuroscience*, 22(18):8117–8132.

McVea, D. and Pearson, K. (2006). Long-lasting memories of obstacles guide leg movements in the walking cat. *The Journal of neuroscience*, 26(4):1175–1178.

Mehler, W. R. (1971). Idea of a new anatomy of the thalamus. *Journal of psychiatric research*, 8(3):203–217.

Merzenich, M. M., Kaas, J., Wall, J., Nelson, R., Sur, M., and Felleman, D. (1983). Topographic reorganization of somatosensory cortical areas 3b and 1 in adult monkeys following restricted deafferentation. *Neuroscience*, 8(1):33–55.

Merzenich, M. M., Nelson, R. J., Stryker, M. P., Cynader, M. S., Schoppmann, A., and Zook, J. M. (1984). Somatosensory cortical map changes following digit amputation in adult monkeys. *Journal of comparative neurology*, 224(4):591–605.

Middleton, F. A. and Strick, P. L. (1994). Anatomical evidence for cerebellar and basal ganglia involvement in higher cognitive function. *Science*, 266(5184):458–461.

Mongillo, G., Barak, O., and Tsodyks, M. (2008). Synaptic theory of working memory. *Science*, 319(5869):1543–1546.

Morimoto, J. and Doya, K. (1998). Reinforcement learning of dynamic motor sequence: Learning to stand up. In *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, pages 1721–1726.

Morimoto, J. and Doya, K. (2001). Acquisition of stand-up behavior by a real robot using hierarchical reinforcement learning. *Robotics and Autonomous Systems*, 36(1):37–51.

Mozer, M. C. (1993). Induction of multiscale temporal structure. *Advances in neural information processing systems*, pages 275–275.

Nakamura, Y. and Nakagawa, M. (2009). Approximation capability of continuous time recurrent neural networks for non-autonomous dynamical systems. In *Artificial Neural Networks–ICANN 2009*, pages 593–602. Springer.

Neychev, V. K., Fan, X., Mitev, V., Hess, E. J., and Jinnah, H. (2008). The basal ganglia and cerebellum interact in the expression of dystonic movement. *Brain*, 131(9):2499–2509.

Ni, Z., Gunraj, C., Kailey, P., Cash, R. F., and Chen, R. (2014). Heterosynaptic modulation of motor cortical plasticity in human. *The Journal of Neuroscience*, 34(21):7314–7321.

O'Doherty, J. P., Dayan, P., Friston, K., Critchley, H., and Dolan, R. J. (2003). Temporal difference models and reward-related learning in the human brain. *Neuron*, 38(2):329–337.

Oh, M. M., Kuo, A. G., Wu, W. W., Sametsky, E. A., and Disterhoft, J. F. (2003). Water-maze learning enhances excitability of ca1 pyramidal neurons. *Journal of neurophysiology*, 90(4):2171–2179.

Oliveri, M., Vicario, C. M., Salerno, S., Koch, G., Turriziani, P., Mangano, R., Chillemi, G., and Caltagirone, C. (2008). Perceiving numbers alters time perception. *Neuroscience letters*, 438(3):308–311.

Olton, D. S. and Samuelson, R. J. (1976). Remembrance of places passed: Spatial memory in rats. *Journal of Experimental Psychology: Animal Behavior Processes*, 2(2):97.

Otani, S., Daniel, H., Roisin, M.-P., and Crepel, F. (2003). Dopaminergic modulation of long-term synaptic plasticity in rat prefrontal neurons. *Cerebral cortex*, 13(11):1251–1256.

Pascual-Leone, A., Amedi, A., Fregni, F., and Merabet, L. B. (2005). The plastic human brain cortex. *Annu. Rev. Neurosci.*, 28:377–401.

Pasemann, F., Hild, M., and Zahedi, K. (2003). So (2)-networks as neural oscillators. In *Computational methods in neural modeling*, pages 144–151. Springer.

Pavlov, I. P. (1927). *Conditioned Reflexes: An Investigation of the Physiological Activity of the Cerebral Cortex.* Oxford University Press: Humphrey Milford.

Pearlmutter, B. A. (1995). Gradient calculations for dynamic recurrent neural networks: A survey. *Neural Networks, IEEE Transactions on*, 6(5):1212–1228.

Pearson, K. and Franklin, R. (1984). Characteristics of leg movements and patterns of coordination in locusts walking on rough terrain. *The International Journal of Robotics Research*, 3(2):101–112.

Percheron, G., Francois, C., Talbi, B., Yelnik, J., and Fenelon, G. (1996). The primate motor thalamus. *Brain research reviews*, 22(2):93–181.

Pierce, W. D. and Cheney, C. D. (2013). *Behavior analysis and learning.* Psychology Press.

Porr, B. and Wörgötter, F. (2006). Strongly improved stability and faster convergence of temporal sequence learning by utilising input correlations only. *Neural computation*, 18(6):1380–1412.

Prescott, T. J., Gonzaelez, F. M., Gurney, K., Humphries, M. D., and Redgrave, P. (2006). A robot model of the basal ganglia: behavior and intrinsic processing. *Neural Networks*, 19(1):31–61.

Prokopenko, M. (2009). Guided self-organization. *HFSP Journal*, 3(5):287–289.

Proville, R. D., Spolidoro, M., Guyon, N., Dugué, G. P., Selimi, F., Isope, P., Popa, D., and Léna, C. (2014). Cerebellum involvement in cortical sensorimotor circuits for the control of voluntary movements. *Nature neuroscience.*

Puig, M. V. and Mille, E. K. (2012). The role of prefrontal dopamine d1 receptors in the neural mechanisms of associative learning. *Neuron*, 74(5):874–886.

Rabinovich, M., Huerta, R., and Laurent, G. (2008). Neuroscience. transient dynamics for neural processing. *Science (New York, NY)*, 321(5885):48–50.

Rall, W. (1969). Time constants and electrotonic length of membrane cylinders and neurons. *Biophysical Journal*, 9(12):1483–1508.

Rao, S. M., Mayer, A. R., and Harrington, D. L. (2001). The evolution of brain activation during temporal processing. *Nature neuroscience*, 4(3):317–323.

Ren, G., Chen, W., Kolodziejski, C., Worgotter, F., Dasgupta, S., and Manoonpong, P. (2012). Multiple chaotic central pattern generators for locomotion generation and leg damage compensation in a hexapod robot. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 2756–2761. IEEE.

Rescorla, R. A. and Solomon, R. L. (1967). Two-process learning theory: Relationships between pavlovian conditioning and instrumental learning. *Psychological review*, 74(3):151–182.

Rodan, A. and Tino, P. (2011). Minimum complexity echo state network. *Neural Networks, IEEE Transactions on*, 22(1):131–144.

Rossler, O. (1979). An equation for hyperchaos. *Physics Letters A*, 71(2):155–157.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive modeling.*

Saar, D., Grossman, Y., and Barkai, E. (1998). Reduced after-hyperpolarization in rat piriform cortex pyramidal neurons is associated with increased learning capability during operant conditioning. *European Journal of Neuroscience*, 10(4):1518–1523.

Sakai, S. T., Stepniewska, I., Qi, H. X., and Kaas, J. H. (2000). Pallidal and cerebellar afferents to pre-supplementary motor area thalamocortical neurons in the owl monkey: a multiple labeling study. *Journal of Comparative Neurology*, 417(2):164–180.

Schrauwen, B., Wardermann, M., Verstraeten, D., Steil, J. J., and Stroobandt, D. (2008). Improving reservoirs using intrinsic plasticity. *Neurocomputing*, 71(7):1159–1171.

Schröder-Schetelig, J., Manoonpong, P., and Wörgötter, F. (2010). Using efference copy and a forward internal model for adaptive biped walking. *Autonomous Robots*, 29(3-4):357–366.

Schultz, W. and Dickinson, A. (2000). Neuronal coding of prediction errors. *Annual review of neuroscience*, 23(1):473–500.

Seung, H. S. (1996). How the brain keeps the eyes still. *Proceedings of the National Academy of Sciences*, 93(23):13339–13344.

Sharpee, T. O., Calhoun, A. J., and Chalasani, S. H. (2014). Information theory of adaptation in neurons, behavior, and mood. *Current opinion in neurobiology*, 25:47–53.

Shettleworth, S. J. (2009). *Cognition, evolution, and behavior.* Oxford University Press.

Siegelmann, H. T. (2010). Complex systems science and brain dynamics. *Frontiers in computational neuroscience*, 4.

Simon, H. (2002). Adaptive filter theory. *Prentice Hall*, 2:478–481.

Skinner, B. F. (1938). *The behavior of organisms: An experimental analysis.* Appleton-Century.

Smith, A., Taylor, E., Lidzba, K., and Rubia, K. (2003). A right hemispheric frontocerebellar network for time discrimination of several hundreds of milliseconds. *Neuroimage*, 20(1):344–350.

Sompolinsky, H., Crisanti, A., and Sommers, H. (1988). Chaos in random neural networks. *Physical Review Letters*, 61(3):259.

Spaan, M. T. (2012). Partially observable markov decision processes. In *Reinforcement Learning*, pages 387–414. Springer.

Sprott, J. C. and Sprott, J. C. (2003). *Chaos and time-series analysis*, volume 69. Oxford University Press Oxford.

Staddon, J. E. (1983). *Adaptive behaviour and learning.* CUP Archive.

Steil, J. J. (2007). Online reservoir adaptation by intrinsic plasticity for backpropagation–decorrelation and echo state learning. *Neural Networks*, 20(3):353–364.

Steingrube, S., Timme, M., Wörgötter, F., and Manoonpong, P. (2010). Self-organized adaptation of a simple neural circuit enables complex robot behaviour. *Nature Physics*, 6(3):224–230.

Stemmler, M. and Koch, C. (1999). How voltage-dependent conductances can adapt to maximize the information encoded by neuronal firing rate. *Nature neuroscience*, 2(6):521–527.

Stepniewska, I., Preuss, T. M., and Kaas, J. H. (1994). Thalamic connections of the primary motor cortex (m1) of owl monkeys. *Journal of Comparative Neurology*, 349(4):558–582.

Stone, M. H. (1948). The generalized weierstrass approximation theorem. *Mathematics Magazine*, 21(5):237–254.

Sturm, J., Plagemann, C., and Burgard, W. (2008). Adaptive body scheme models for robust robotic manipulation. In *Robotics: Science and systems*.

Sugrue, L. P., Corrado, G. S., and Newsome, W. T. (2004). Matching behavior and the representation of value in the parietal cortex. *science*, 304(5678):1782–1787.

Sul, J. H., Jo, S., Lee, D., and Jung, M. W. (2011). Role of rodent secondary motor cortex in value-based action selection. *Nature neuroscience*, 14(9):1202–1208.

Suri, R. E. and Schultz, W. (2001). Temporal difference model reproduces anticipatory neural activity. *Neural Computation*, 13(4):841–862.

Sussillo, D. (2014). Neural circuits as computational dynamical systems. *Current opinion in neurobiology*, 25:156–163.

Sussillo, D. and Abbott, L. (2012). Transferring learning from external to internal weights in echo-state networks with sparse connectivity. *PloS one*, 7(5):e37372.

Sussillo, D. and Abbott, L. F. (2009). Generating coherent patterns of activity from chaotic neural networks. *Neuron*, 63(4):544–557.

Sussillo, D., Nuyujukian, P., Fan, J. M., Kao, J. C., Stavisky, S. D., Ryu, S., and Shenoy, K. (2012). A recurrent neural network for closed-loop intracortical brain–machine interface decoders. *Journal of neural engineering*, 9(2):026027.

Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement learning: An Introduction.* MIT Press.

Taatgen, N. A., Van Rijn, H., and Anderson, J. (2007). An integrated theory of prospective time interval estimation: the role of cognition, attention, and learning. *Psychological Review*, 114:577.

Takikawa, Y., Kawagoe, R., and Hikosaka, O. (2004). A possible role of midbrain dopamine neurons in short-and long-term adaptation of saccades to position-reward mapping. *Journal of neurophysiology*, 92(4):2520–2529.

Tetzlaff, C., Dasgupta, S., and Wörgötter, F. (2014). The association between cell assemblies and transient dynamics. *BMC Neuroscience*, 15(Suppl 1):P10.

Tetzlaff, C., Kolodziejski, C., Markelic, I., and Wörgötter, F. (2012a). Time scales of memory, learning, and plasticity. *Biol. Cybern.*, 106(11):715–726.

Tetzlaff, C., Kolodziejski, C., Timme, M., and Wörgötter, F. (2012b). Analysis of synaptic scaling in combination with hebbian plasticity in several simple networks. *Front. Comput. Neurosci.*, 6:36.

Thompson, R. and Steinmetz, J. (2009). The role of the cerebellum in classical conditioning of discrete behavioral responses. *Neuroscience*, 162(3):732–755.

Tolman, E. C. (1932). *Purposive behavior in animals and men.* Berkeley: University of California Press.

Tolman, E. C. and Honzik, C. H. (1930). Introduction and removal of reward, and maze performance in rats. *University of California Publications in Psychology.*

Toutounji, H. and Pipa, G. (2014). Spatiotemporal computations of an excitable and plastic brain: neuronal plasticity leads to noise-robust and noise-constructive computations. *PLoS computational biology*, 10(3):e1003512.

Triefenbach, F., Jalalvand, A., Schrauwen, B., and Martens, J.-P. (2010). Phoneme recognition with large hierarchical reservoirs. In *Advances in neural information processing systems*, pages 2307–2315.

Triesch, J. (2007). Synergies between intrinsic and synaptic plasticity mechanisms. *Neural Comput.*, 19:885–909.

Turrigiano, G. G., Abbott, L. F., and Marder, E. (1994). Activity dependent changes in the intrinsic properties of cultured neurons. *Science*, 264:974–977.

Turrigiano, G. G., Leslie, K. R., Desai, N. S., Rutherford, L. C., and Nelson, S. B. (1998). Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature*, 391:892–896.

Turrigiano, G. G. and Nelson, S. B. (2004). Homeostatic plasticity in the developing nervous system. *Nat. Rev. Neurosci.*, 5:97–107.

van Vreeswijk, C. and Sompolinsky, H. (1996). Chaos in neuronal networks with balanced excitatory and inhibitory activity. *Science*, 274(5293):1724–1726.

van Welie, I., van Hooft, J. A., and Wadman, W. J. (2004). Homeostatic scaling of neuronal excitability by synaptic modulation of somatic hyperpolarization-activated ih channels. *Proceedings of the National Academy of Sciences of the United States of America*, 101(14):5123–5128.

Varela, C. (2014). Thalamic neuromodulation and its implications for executive networks. *Frontiers in Neural Circuits*, 8(69).

Verschure, P. F. and Mintz, M. (2001). A real-time model of the cerebellar circuitry underlying classical conditioning: A combined simulation and robotics study. *Neurocomputing*, 38:1019–1024.

Verstraeten, D., Schrauwen, B., dâĂŹHaene, M., and Stroobandt, D. (2007). An experimental unification of reservoir computing methods. *Neural Networks*, 20(3):391–403.

Vincent, B. T., Baddeley, R. J., Troscianko, T., and Gilchrist, I. D. (2005). Is the early visual system optimised to be energy efficient? *Network: Computation in Neural Systems*, 16(2-3):175–190.

Vitureira, N., Letellier, M., and Goda, Y. (2012). Homeostatic synaptic plasticity: from single synapses to neural circuits. *Current opinion in neurobiology*, 22(3):516–521.

Wang, X.-J. (2001). Synaptic reverberation underlying mnemonic persistent activity. *Trends in neurosciences*, 24(8):455–463.

Watson, J. T., Ritzmann, R. E., Zill, S. N., and Pollack, A. J. (2002). Control of obstacle climbing in the cockroach, blaberus discoidalis. i. kinematics. *Journal of Comparative Physiology A*, 188(1):39–53.

Webb, B. (2004). Neural mechanisms for prediction: do insects have forward models? *Trends in neurosciences*, 27(5):278–282.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560.

Wibral, M., Lizier, J. T., Vögler, S., Priesemann, V., and Galuske, R. (2014a). Local active information storage as a tool to understand distributed neural information processing. *Frontiers in neuroinformatics*, 8.

Wibral, M., Vicente, R., and Lizier, J. T. (2014b). *Directed Information Measures in Neuroscience*. Springer.

Williams, R. J. and Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280.

Wilson, H. R. and Cowan, J. D. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical journal*, 12(1):1–24.

Wolpert, D. M., Ghahramani, Z., and Jordan, M. I. (1995). An internal model for sensorimotor integration. *Science*, pages 1880–1880.

Wolpert, D. M., Miall, R. C., and Kawato, M. (1998). Internal models in the cerebellum. *Trends in cognitive sciences*, 2(9):338–347.

Woodruff-Pak, D. S. and Disterhoft, J. F. (2008). Where is the trace in trace conditioning? *Trends in neurosciences*, 31(2):105–112.

Wörgötter, F. and Porr, B. (2005). Temporal sequence learning, prediction, and control: a review of different models and their relation to biological mechanisms. *Neural Computatio*, 17(2):245–319.

Wyffels, F. and Schrauwen, B. (2010). A comparative study of reservoir computing strategies for monthly time series prediction. *Neurocomputing*, 73(10):1958–1964.

Wyffels, F., Schrauwen, B., Verstraeten, D., and Stroobandt, D. (2008). Band-pass reservoir computing. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 3204–3209. IEEE.

Xue, Y., Yang, L., and Haykin, S. (2007). Decoupled echo state networks with lateral inhibition. *Neural Networks*, 20(3):365–376.

Yamashita, Y. and Tani, J. (2008). Emergence of functional hierarchy in a multiple timescale neural network model: a humanoid robot experiment. *PLoS computational biology*, 4(11):e1000220.

Yeo, C. H. and Hesslow, G. (1998). Cerebellum and conditioned reflexes. *Trends in cognitive sciences*, 2(9):322–330.

Zenker, S., Aksoy, E. E., Goldschmidt, D., Worgotter, F., and Manoonpong, P. (2013). Visual terrain classification for selecting energy efficient gaits of a hexapod robot. In *Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on*, pages 577–584. IEEE.

Zhang, W. and Linden, D. J. (2003). The other side of the engram: experience-driven changes in neuronal intrinsic excitability. *Nat. Rev. Neurosci.*, 4:886–900.

Zill, S., Schmitz, J., and Büschges, A. (2004). Load sensing and control of posture and locomotion. *Arthropod structure & development*, 33(3):273–286.

APPENDIX

A.1 Information Theoretic Measures

In this section we present a basic overview of information theoretic measures relevant to this thesis. All the information theoretic measures used in this thesis (see chapter 2) were implemented using modifications to the Java Information Dynamics Toolkit (Lizier, 2014) and used as Java wrapper called from within Matlab and C++.

The fundamental quantity of information theory is the **Shannon entropy**, representing the average uncertainty associated with the measurement x of a random variable X, calculated as:

$$H(X) = -\sum_{x} p(x) log_2 p(x). \tag{A.1}$$

The **joint entropy** of two random variables X and Y is a generalization to quantify the uncertainty of their joint distribution:

$$H(X,Y) = -\sum_{x}\sum_{y} p(x,y) log_2 p(x,y). \tag{A.2}$$

The **conditional entropy** of X given Y is the expected uncertainty that remains about x when y is known:

$$H(X|Y) = -\sum_{x}\sum_{y} p(x,y) log_2 p(x|y). \tag{A.3}$$

In general the previous quantities are related to each other as follows:

$$H(X,Y) = H(X) + H(Y|X). \tag{A.4}$$

The **mutual information** between X and Y measures the average reduction in uncertainty about x that results from learning the value of y, or vice versa:

$$\begin{aligned} I(X;Y) &= \sum_x \sum_y p(x,y) log_2 \frac{p(x|y)}{p(x)}, \\ &= H(X) - H(X|Y). \end{aligned} \tag{A.5}$$

The **conditional mutual information** between X and Y given Z is the mutual information between X and Y when Z is known:

$$\begin{aligned} I(X;Y|Z) &= \sum_x \sum_y \sum_z p(x,y,z) log_2 \frac{p(x|y,z)}{p(x|z)}, \\ &= \sum_x \sum_y \sum_z p(x,y,z) log_2 \frac{p(x,y,z)p(z)}{p(x,z)p(y,z)}, \\ &= H(X|Z) - H(X|Y,Z). \end{aligned} \tag{A.6}$$

In all the above cases without taking the sum of x, y and z, and taking only the quantities inside the log_2 along with any sign in front of the equation, one can calculate the local entropy, local conditional entropy and local mutual information for events x_i and y_i. This is the way we calculate the local active information storage values in equation 2.17.

The **Kullback-Leibler divergence** (KL-divergence) is a non-symmetric measure that can be used to compare (difference between) two probability distributions P and Q. In general the KL-divergence of Q from P is the information lost when Q is used to approximate P. For discrete probability distributions it can be calculated as:

$$D_{KL} = \sum_i p_x log \frac{P(x)}{Q(x)}. \tag{A.7}$$

Similarly, in case of continuous probability distributions, it can be computed as:

$$D_{KL} = \int_{-\infty}^{\infty} p(x) log \frac{p(x)}{q(x)}. \tag{A.8}$$

The KL-divergence value is always positive and only equal to zero if the two probability distributions are equal almost everywhere.

Here, in order to calculate the relevant measures for the continuous random variable x_i of the reservoir neuron activation states, we simply discretise the data using bins and apply the above mentioned measures and the measures for local and average active information storage, as introduced in section 2.2.2.

A.2 Estimating Dynamics with Largest Lyapunov Exponent

It is possible to determine whether a dynamical system has ordered or chaotic dynamics by looking at the average sensitivity to perturbations of its initial conditions (Kantz, 1994). If two systems who are otherwise equal, are in the ordered phase, then small differences in the initial conditions of these systems should eventually die out. However if they are in the chaotic state, then these small differences will persist or get amplified in time. The exponential divergence of two trajectories of a dynamical system in state space with very small initial separation, can be measured using the Lyapunov (characteristic) exponent (LE). In general, a whole spectrum of Lyapunov exponents are defined, however, the rate of divergence is dominated by the largest Lyapunov exponent (LLE). Mathematically it can be defined as:

$$\lambda = \lim_{k \to \infty} \frac{1}{n} \mathtt{ln} \left(\frac{\gamma_n}{\gamma_0} \right). \tag{A.9}$$

where, γ_0 is the initial separation (distance) between the perturbed and the unperturbed trajectory, and γ_n is the distance at time n. The largest Lyapunov exponent, $\lambda < 0$ in case of ordered, sub-critical systems; $\lambda > 0$ in case of chaotic systems and $\lambda \approx 0$ is the region where phase transitions occur and is thus called the critical point or the so called *'edge of chaos'*.

In the case of the reservoir networks, the recurrent layer neuron activity is basically a time series data. As such, here we estimate LLE using a finite time estimation technique based on (Sprott and Sprott, 2003). In order to estimate λ, we simulated two identical versions of the reservoir network (static or SARN) for a period of 1000 time steps ($\Delta t = 1.0$). After this initial transient the following procedure was followed:

1. A small perturbation is introduced in one of the units i of one network, leaving the other intact. As a result there occurs a separation of the state of the perturbed network ($\mathbf{x}'$) from the state of the unperturbed network ($\mathbf{x}$), by a distance γ_0 (here we used an initial separation of 10^{-12}).
2. **Simulation step**: Advance the simulation by one time step (n) calculate the resulting distance or separation (Euclidean norm) between the states as $\gamma_n = \|\mathbf{x}(n) - \mathbf{x}'(n)\|$.
3. **Normalization step**: Reset the state of the perturbed network $\mathbf{x}'$ to $\mathbf{x}(n) + (\gamma_0/\gamma_n)(\mathbf{x}'(n) - \mathbf{x}(n))$, such that the two trajectories remain close in order to avoid numerical overflows.
4. Repeat steps 2 and 3 for 10000 times. The largest Lyapunov exponent for the trajectory of each reservoir neuron is then calculated as the time average of the logarithm of the distances along the trajectory, $\lambda_i = \langle \mathtt{ln}(\gamma_n/\gamma_0) \rangle_n$.

Given, a reservoir of size N neurons. We calculate λ_i for each of the N neurons. Then the finite estimate of the largest Lyapunov exponent for the network is obtained by averaging over all the neurons, i.e. LLE $(\lambda) = \langle \lambda_i \rangle_i$.

A.3 Modular Neural Locomotion Control

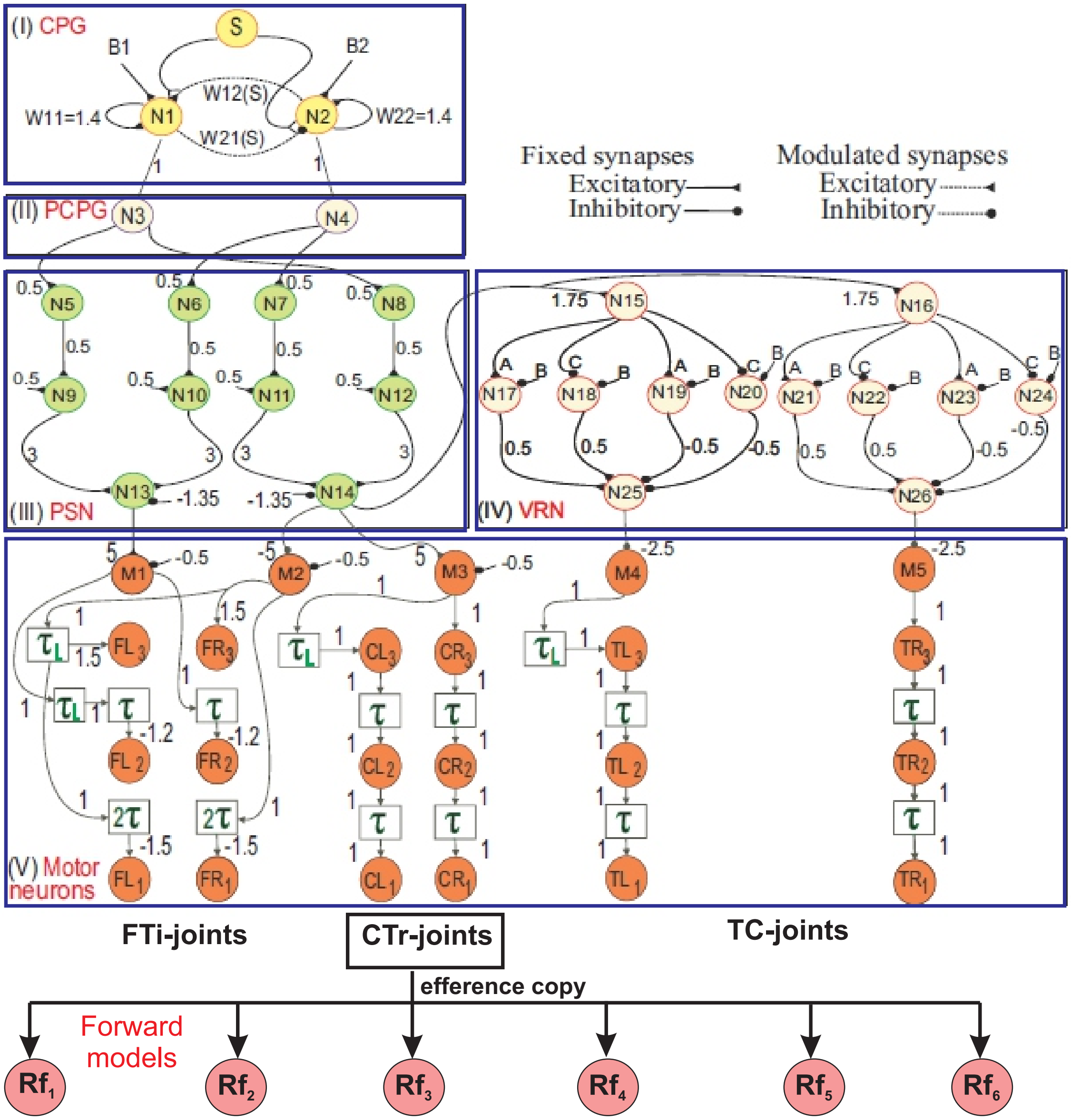


Figure A.1: **Main wiring diagram of the neural locomotion control with central pattern generator** Single CPG-based control applied to AMOSII for locomotion. CPG's outputs are projected to PCPG (CPG post processing unit) which translate them into ascending and descending slopes, then these slops will be fed to the PSN (phase shift network) component. The outputs of the PSN are projected to the F(R,L) and C(R,L) motor neurons (i.e the FTi and CTr joints of the robot) through delay lines, as well as to the VRN (velocity regulating network). The VRN's outputs are projected to the T(R,L) motor neurons (TC joints) through delay lines. The CT joint signals are then used as efference copies that feed as input to each of the six reservoir forward models Rf_1 to Rf_6. Adapted and modified from (Manoonpong et al., 2013b)

A.4 Neuromodulatory combined learning (additional experimental results)

Dopaminergic neurons are primarily believed to encode a reward prediction error (RPE) signal (Schultz and Dickinson, 2000). Although, recent experimental evidences have shown that a subset of the VTA dopaminergic neurons can directly encode the reward signal, most of them still follow the canonical RPE coding (Cohen et al., 2012). In the context of the actor-critic reservoir model of the basal ganglia, the temporal difference error (TD-error) is considered as the prediction error signal output of the dopaminergic neurons (Suri and Schultz, 2001). As such, in order to test the stability and efficiency of the reward modulated heterosynaptic (RMHP) combined learning rule while using the TD-error ($\delta(t)$) as the neuromodulatory signal at the motor thalamic junction instead of the instantaneous reward signal $R(t)$, we modified Eq. 5.2 and Eq. 5.3 as follows:

$$\Delta\xi_{ico}(t) = \eta\delta(t)[o_{ico}(t) - \bar{o}_{ico}(t)]o_{ac}(t), \tag{A.10}$$

$$\Delta\xi_{ac}(t) = \eta\delta(t)[o_{ac}(t) - \bar{o}_{ac}(t)]o_{ico}(t). \tag{A.11}$$

Here, the TD-error signal($\delta(t)$) is calculated as part of the reservoir critic network and updated based on the current reward and the estimated sum of future rewards ($\hat{v}(t)$) at every time time step as follows:

$$\delta(t) = R(t) + \gamma\hat{v}(t) - \hat{v}(t-1). \tag{A.12}$$

We tested the performance of the modified learning rule on the foraging scenario with a single obstacle (Chapter 5, Fig. 5.9 (B)) with no changes to the experimental setup. 20 runs were carried out with the original RMHP rule (direct reward signal modulation) and the modified RMPH rule (TD-error modulation). As observed in Fig. A.2 (A)), the robot was successfully able to complete the task with only a single failure, achieving a performance rate of 95% in both cases. Fig. A.2 (B), shows the average learning time needed to learn the task under both conditions. The TD-error based learning rule took negligibly longer time to converge to a solution (57 trials) in comparison to the instantaneous reward-based learning rule (54 trials). This behavior can be attributed to the fact that, the TD-error signal is updated continuously resulting in the ICO learner(ξ_{ico}) and the actor-critic learner (ξ_{ac}) weights changing all the time. This is avoided in the direct reward based RMHP rule, since the reward signal $R(t)$ is active only within the positive or negative reward zone and zero otherwise. As a result, any initial wrong estimates by the critic do not effect the combined learning weights, substantially.
Over all our results prove that the RMHP combined learning rule works stably with similar

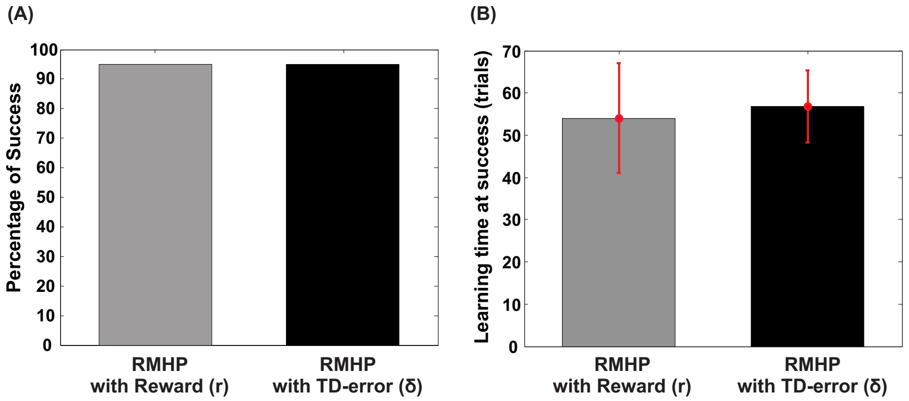


Figure A.2: **Comparison of performance of RMHP modulated adaptive comninatorial learning system with direct reward (original) and TD-error modulation, in the single obstacle foraging case**. **(A)** Percentage of success measured over 20 experiments. **(B)** Average learning time (trials needed to successfully complete the task, calculated over 20 experiments (error bars indicate standard deviation with 98% confidence intervals). In both cases the grey bars represent the performance for task of learning to reach the green goal with the original RMHP rule , while black bars represent the performance in the same task using the TD-error modulated RMHP rule.

levels of performance, independent of the choice of the instantaneous reward or the temporal difference error as the modulatory signal. However, in this work we have only tested goal-directed decision making scenarios. In other learning scenarios like dynamic motion control (Morimoto and Doya, 2001) there may be differences in performance for the two variants of the RMHP rule. This would require further analysis under various environmental conditions and goes beyond the scope of the current paper. In essence the current scheme of RMHP (in both variants of direct reward and TD-error modulations) provides an effective and efficient mechanism to combine the reward learning and correlation learning systems of the basal ganglia and the cerebellum brain structures, respectively.

Parameter description	Value
Time constant of the reservoir critic (τ)	1s
Reservoir critic size (N - neurons)	100
Forgetting factor (γ)	0.98
Critic scaling factor (g)	1.2
Critic bias input (b)	0.001
Auto-correlation matrix constant (δ_c)	10^{-2}
Exploration scale factor (Ω)	5
Maximum value function (v_{max})	50
Minimum value function (v_{min})	-50
Learning rate of actor (τ_a)	0.005
Critic input weights (W^{in})	fixed Uniform [-0.5,0.5]
Critic recurrent weights (W^{rec})	fixed Normal (0, $g^2/\sqrt{p_c N}$)
Recurrent connection probability (p_c)	0.1
Critic output weights (W^{out})	plastic
Initialization of actor weights (w_{μ_G} and w_{μ_B})	0.0
Initialization of actor weights (w_{IR_1} and w_{IR_2})	0.5
Number of inputs (K)	4
Number of output	1

Table A.1: Parameters of the actor-critic reinforcement learning network

Parameter description	Value
Strength of reflex signal (ρ_0)	1.0
Learning rate (μ)	0.001
Initialization of input weights (ρ_{μ_G} and ρ_{μ_B})	0.0
Number of inputs (K)	2
Number of output	1

Table A.2: Parameters of the input correlation learning (ICO) network

Parameter description	Value
Initialization of individual learner weights (ξ_{ico} and ξ_{ac})	0.5
Learning rate (η)	0.0005

Table A.3: Parameters of the combinatorial learner (RMHP rule)

Algorithm 1 : Adaptive Neural combinatorial learning algorithm

```
Input:
    • Actor-critic RL: input stimuli vector $u_{1,2,3,4} = \mu_G, \mu_B, IR_1, IR_2$
    • ICO learning: input stimuli vector $x_{1,2} = \mu_G, \mu_B$
Initialization:
    • ICO weights: $\rho_{\mu_G}$, $\rho_{\mu_B} = 0.0$; $\rho_0 = 1.0$ (reflex signal strength)
    • Actor weights: $w_{\mu_G}$, $w_{\mu_B} = 0.0$; $w_{IR_1}, w_{IR_1} = 0.5$
    • RMHP combined learner weights: $\xi_{ico}, \xi_{ac} = 0.5$
    • exploration noise $\epsilon$: approximately normal distribution calculated as sum of 'n' i.i.d r.v ∈ U(0,1)
Observe reflex signal $x_0$ and the sensory signals $x_{1,2}(t)$ and $u_{1,2,3,4}(t)$
while (i < max time steps) do
Execution:
    • $o_{ico}(t) \leftarrow \rho_0 x_0(t) + \sum_{j=1}^{K} \rho_j(t) x_j(t)$
    • $o_{ac}(t) \leftarrow \epsilon(t) + \sum_{i=1}^{K} w_i(t) u_i(t)$
    • $o_{com}(t) \leftarrow \xi_{ico} o_{ico}(t) + \xi_{ac} o_{ac}(t)$
Perform action
Observe new sensory states $x'(t)$, $u'(t)$ and new reflex signal $x'_0(t)$
Update the reward signal $R(t)$ :
    if robot is within the green reward zone $(D_G < 0.2)$ then
    $R(t) = +1$
    end if
    if robot is within the blue reward zone $(D_B < 0.2)$ then
    $R(t) = -1$
    end if
    if $IR_1 > 1.0$ or $IR_2 > 1.0$ then
    $R(t) = -1$
    end if
Update value prediction from critic:
    • $\tau \dot{\mathbf{x}}(t) \leftarrow -\mathbf{x}(t) + g\mathbf{W}^{rec}\mathbf{r}(t) + \mathbf{W}^{in}\mathbf{u}(t) + \mathbf{b}$
    • $\hat{v}(t) \leftarrow \mathtt{tanh}(\mathbf{W}^{out}\mathbf{r}(t))$
Update exploration noise: $\epsilon(t) \leftarrow \Omega\sigma(t) \cdot \mathtt{min}\Big[0.5, \mathtt{max}\Big(0, \frac{v_{max} - \hat{v}(t)}{v_{max} - v_{min}}\Big)\Big]$
Calculate temporal difference (prediction) error : $\delta(t) \leftarrow R(t) + \gamma\hat{v}(t) - \hat{v}(t-1)$.
Update all synaptic weights:
    • ICO weights : $\frac{d}{dt}\rho_j(t) \leftarrow \mu x_j(t) \frac{d}{dt} x_0(t)$
    • Critic weights: $\mathbf{W}^{out}(t) \leftarrow \mathbf{W}^{out}(t-1) + \delta(t)\mathbf{P}(t)\mathbf{r}(t)$
    • Actor weights: $\Delta w_i(t) \leftarrow \tau_a \delta(t) u_i(t) \epsilon(t)$
    • RMHP weights: $\Delta\xi_{ico}(t) \leftarrow \eta R(t)(o_{ico}(t) - \bar{o}_{ico}(t)) o_{ac}(t)$ ; $\Delta\xi_{ac}(t) \leftarrow \eta R(t)(o_{ac}(t) - \bar{o}_{ac}(t)) o_{ico}(t)$
$i = i + 1$
```

ACADEMIC CURRICULUM VITAE

Personal Details:

Family name:	Dasgupta
First name:	Sakyasingha
Nationality:	Indian
Date of birth:	08/28/1985
Place of birth:	Assam, India

Education:

08/2004-07/2008:	B.E in Computer Science and Engineering, P.E.S Institute of Technology, Bangalore, India.
09/2009-11/2010:	M.Sc. in Artificial Intelligence, The University of Edinburgh, Edinburgh, United Kingdom.
02/2012-now:	PhD-student at the Göttingen Graduate School for Neurosciences, Biophysics, and Molecular Biosciences (GGNB), Georg-August University, Göttingen, Germany.

Awards:

2012:	Best student paper award, at Engineering Applications of Neural Networks conf. (EANN 2012), London, U.K. awarded by the International Neural Network Society (INNS).
05/2012-06/2014:	Scholarship from International Max Planck Research School Göttingen, Germany (state of Lower Saxony award).

Journal Publications:

Ren, G., Chen, W., **Dasgupta, S.**, Kolodziejski, C., Wörgötter, F., & Manoonpong, P. (2015). Multiple chaotic central pattern generators with learning for legged locomotion and malfunction compensation. *Information Sciences*, 294, 666-682, doi:10.1016/j.ins.2014.05.001.

Dasgupta, S., Wörgötter, F., & Manoonpong, P. (2014). Neuromodulatory adaptive combination of correlation-based learning in cerebellum and reward-based learning in basal ganglia for goal-directed behavior control. *Frontiers in Neural Circuits*, 8:126, doi: 10.3389/fncir.2014.00126.

Dasgupta, S., Wörgötter, F., & Manoonpong, P. (2013). Information dynamics based self-adaptive reservoir for delay temporal memory tasks. *Evolving Systems*, 4(4), 235-249, doi: 10.1007/s12530-013-9080-y.

Conference Publications:

*Manoonpong, P., ***Dasgupta, S.**, Goldschimdt, D., & Wörgötter, F. (2014). Reservoir-based online adaptive forward models with neural control for complex locomotion in a hexapod robot. *Neural Networks (IJCNN), 2014 International Joint Conference on*, (pp.3295,3302), 6-11 July 2014, doi: 10.1109/IJCNN.2014.6889405. *equal contribution

Dasgupta, S., Wörgötter, F., Morimoto, J., & Manoonpong, P. (2013). Neural combinatorial learning of goal-directed behavior with reservoir critic and reward modulated hebbian plasticity. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on* (pp. 993-1000), doi: 10.1109/SMC.2013.174. IEEE.

Conference Publications (contd.):

Ren, G., Chen, W., Kolodziejski, C., Wörgötter, F., **Dasgupta, S.**, & Manoonpong, P. (2012). Multiple chaotic central pattern generators for locomotion generation and leg damage compensation in a hexapod robot. *Intelligent Robots and Systems (IROS), 2012 IEEE International Conference on*, (pp. 2756-2761), doi:10.1109/IROS.2012.6385573.

Dasgupta, S., Herrmann, M.J. (2011). Critical dynamics in homeostatic memory networks. *Nature Precedings*, doi: 10.1038/npre.2011.5829.1.

Book Chapters:

Zeidan, B., **Dasgupta, S.**, Wörgötter, F., & Manoonpong, P. (2014). Adaptive Landmark-Based Navigation System Using Learning Techniques. *From Animals to Animats 13, 13th International Conference on Simulation of Adaptive Behavior, (SAB) 2014*, 8575, 121-131, doi: 10.1007/978-3-319-08864-8_12.

Dasgupta, S., Wörgötter, F., & Manoonpong, P. (2012). Information theoretic self-organised adaptation in reservoirs for temporal memory tasks. In *Engineering Applications of Neural Networks*, (pp. 31-40, 311), doi: 10.1007/978-3-642-32909-8_4. Springer Berlin Heidelberg.

Conference & Workshop Abstracts (peer-reviewed):

Dasgupta, S., Manoonpong, P., & Wörgötter, F. (2014). Reservoir of neurons with adaptive time constants: a hybrid model for robust motor-sensory temporal processing. *BMC Neuroscience*, 15(Suppl 1):P9, doi:10.1186/1471-2202-15-S1-P9.

Tetzlaff, C., **Dasgupta, S.**, & Wörgötter, F. (2014). The association between cell assemblies and transient dynamics. *BMC Neuroscience*, 15(Suppl 1):P10, doi: 10.1186/1471-2202-15-S1-P10.

Dasgupta, S., Wörgötter, F., & Manoonpong, P. (2013). Population clock models and delayed temporal memory: An information theoretic approach. *10th Meeting of the German Neuroscience Society (Goettingen Neurobiology Conference), 13-16 March*, T25-7D.

DECLARATION OF ORIGINALITY

"I hereby declare that:

1. The opportunity to work on this doctoral thesis project was not arranged commercially. Especially, I did not engage any organization which searches for doctoral thesis supervisors or which will entirely or partly carry out my examination duties against payment;
2. I have only accepted and will only accept the assistance of third parties in so far as it is scientifically justifiable and acceptable in regards to the examination regulations. Especially, all parts of the dissertation will be written by myself; I have not accepted and will not accept impermissible help from other parties neither for money nor for free;
3. I will observe the Statue of the Georg-August-University Göttingen for ensuring good scientific practice;
4. I have not applied for corresponding doctoral degree procedures at any other university in Germany or abroad; the submitted doctoral thesis or parts thereof were not used in another doctoral degree procedure.

I am aware that incorrect information precludes the admission to doctoral studies or will later on lead to the discontinuation of the doctoral degree procedures or to the revocation of the doctoral degree."

Göttingen, 19/12/2014
Place, Date

Sakyasingha Dasgupta