ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE
ENGINEERING AND TECHNOLOGY

NEUROCOMPUTATIONAL MODELS FOR ACTION SELECTION AND THEIR IMPLEMENTATION ON ROBOTS

M.Sc. THESIS

Emeç ERÇELİK

Electronics and Communication Department

Electronics Engineering Programme

DECEMBER 2015

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE

ENGINEERING AND TECHNOLOGY

NEUROCOMPUTATIONAL MODELS FOR ACTION SELECTION AND THEIR IMPLEMENTATION ON ROBOTS

M.Sc. THESIS

Emeç ERÇELİK
(504131206)

Electronics and Communication Department

Electronics Engineering Programme

Thesis Advisor: Prof. Dr. Neslihan Serap Şengör

DECEMBER 2015

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

HAREKET SEÇİMİNE İLİŞKİN BEYİN ESİNLENMELİ HESAPLAMALI MODELLER VE ROBOTLAR ÜSTÜNDE GERÇEKLEME

YÜKSEK LİSANS TEZİ

Emeç ERÇELİK
(504131206)

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Elektronik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Neslihan Serap Şengör

ARALIK 2015

Emeç ERÇELİK, a M.Sc. student of ITU Graduate School of Science Engineering and Technology 504131206 successfully defended the thesis entitled **"NEUROCOMPUTATIONAL MODELS FOR ACTION SELECTION AND THEIR IMPLEMENTATION ON ROBOTS"**, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. Neslihan Serap Şengör** Istanbul Technical University	....
Jury Members :	**Yrd. Doç. Dr. Sanem Sarıel** Istanbul Technical University	....
	Doç. Dr. Erhan Öztop Özyeğin University	....

Date of Submission : **27 November 2015**
Date of Defense : **22 December 2015**

To my family and my love Büşra,

FOREWORD

First of all, I would like to thank to my thesis advisor, Prof. Dr. Neslihan Serap Şengör for her valuable guidance and support. She was the one that listens and advices me anytime with patience whatever the topic is and none of my studies would be done without her. I also feel great honour to be a member of ITU Neuroscience Modelling and Research Group and would like to thank to all members of this group. Through my thesis study, I was sure to find an answer when I asked questions to Berat Denizdurduran and Rahmi Elibol. I am thankful to them for being there to answer anytime and great advices not only for my studies but also for life. In addition, this is a nice opportunity to thank to Damien Jade Duff for his valuable ideas on my thesis and his support that I am doing well.

I am thankful to Assoc. Prof. Dr. Berna Örs for her always open door to talk. I am also thankful to Ramazan Yeniçeri for sharing his endless energy during my works.

I am also grateful to my friend Baki Berkay Yılmaz for the long talks by which we encouraged ourselves for the future works. I would like to take this opportunity to thank to Dursun Çağdaş Akalın for his priceless influence to my life.

I would like to thank to all my family, especially my parents and my brother for always supporting and trusting me on my steps.

Lastly, I would like to thank to my love Büşra. Her patience and support never ended during my studies. She has always endured my "I have to study" words and accepted to endure me for a lifetime.

December 2015 Emeç ERÇELİK

TABLE OF CONTENTS

Page

ABBREVIATIONS

BG	: Basal ganglia
BTC	: Basal ganglia-thalamus-cortex
Ctx	: Cortex
Thl	: Thalamus
Stn	: Subthalamic Nucleus
GPi	: Globus pallidus internel
GPe	: Globus pallidus external
Ch	: Channel
DA	: Dopamine
TDL	: Temporal difference learning
STDP	: Spike timing dependent plasticity
BTK	: Bazal ganglia-talamus-korteks
EEC	: Embodied, embedded cognition
RL	: Reinforcement Learning
PD	: Parkinson's disease
TS	: Tourette syndrome
HD	: Huntington's disease
SNN	: Spiking Neural Network
BT	: Basal ganglia-thalamus
LTP	: Long term potentiation
LTD	: Long term depression

LIST OF TABLES

LIST OF FIGURES

NEUROCOMPUTATIONAL MODELS FOR ACTION SELECTION AND THEIR IMPLEMENTATION ON ROBOTS

SUMMARY

Computational models of neural circuits enhances our comprehension of brain functions. In addition to the simulation of the models which helps to anticipate the cognitive processes, embodiment of these models is essential. Such embodiment would provide necessary setting to explain neural functioning ongoing in real environments under oncoming sensory information. Also, these studies boost the work on intelligent systems by providing new approaches and techniques for the implementation of intelligent methods. Even though studies pursued in neuroscience can be considered as being in inception period, the embodiment of models done since now, reached the pre-results faster than the animal experiments. So, computational neuroscience is promising to lead further understanding of cognitive processes and design of related experiments.

In this thesis, the main aim is to show the embodiment of computational models is possible for different scales of computational models that are biologically meaningful. Still another aim is also show that the implemented models are meaningful to get inference about the behavioural processes of brain circuits. For the embodiment part of the thesis, the Darwin-OP humanoid robot platform is utilized mainly, while the Bioloid robot environment is also considered to get some of the results.

To realize the aims mentioned above, a temporal sequence task related to action selection is utilized. In this task, we investigated the associations between the sensory stimuli and desired actions, and also the mechanism by which reassociations result in development of new associations over the built up ones. Since the action selection is basically linked to the basal ganglia, thalamus and cortex (BTC) circuit in the brain, the BTC structures of brain are modeled in different scales to realize the considered task. The proposed models are the mass model approach of nonlinear dynamical system modeling and point neuron based models. In order to ensure the second aim, the mass model approach is deeply investigated to obtain some of the biological results with this model. Afterwards, the cortex part of the model is redesigned using point neurons to realize a more realistically plausible model.

In addition to realization of BTC circuit, learning process is considered to make associations in order to select the right action in long term encountering. So, the temporal difference learning (TDL) is utilized to ensure the biological plausibility. Thus, reinforcement learning method is utilized for the learning part of the mass model. Although, TDL ensures the biological plausibility, it is a rule based model anyway. So, though it is possible to merge TDL with point neuron based models, spike timing dependent plasticity (STDP), which is more convenient from the biological aspect, is utilized for the learning part of the point neuron based action selection model.

The investigation of the mass model shows that it is possible to obtain meaningful results from the biological aspect using the computational models. Another result of this thesis is that it is possible to implement different scales of computational models for cognitive processes into robots and run in real time applications. So, the results show that, using these computational models to realize complex tasks in future will infer further results. As a result, this thesis is a step to reach evaluating such cognitive models for the complex tasks in real environment and also, that it is possible in near future.

HAREKET SEÇİMİNE İLİŞKİN BEYİN ESİNLENMELİ HESAPLAMALI MODELLER VE ROBOTLAR ÜSTÜNDE GERÇEKLEME

ÖZET

Bu tezin bir amacı, merkezi sinir sistemindeki süreçlerden yararlanılarak oluşturulmuş hesaplamalı modeller ile fonksiyonel açıdan beyni incelemek ve bilişsel süreçler ile davranışsal süreçleri açıklamada bu hesaplamalı modellerin faydalı olabileceğini göstermektir. Diğer bir amacı ise bu hesaplamalı modellerin robotlar üzerinde gerçeklenerek somutlaştırılabileceğini ve karmaşık ve bilişsel süreçlere ilişkin görevleri gerçeklemede kullanılabilineceğini göstermektir.

Ele aldığımız hesaplamalı modellerin bir donanım üzerinde de işlevsel olabileceğini ve donanım aracılığı ile çevresel uyaranların algılanıp, hesaplamalı model aracılığı ile değerlendirilebileceğini göstermek amacıyla ilk olarak Bioloid robotu kullanılmıştır. Bioloid, kullanılan hesaplamalı modellerin, hareket özelliği olan bir donanım ile birlikte kullanılmasını sağlayan bir ortam olsa bile, karmaşık modelleri çalıştırmak için işlemci gücü açısından yetersiz kalmıştır. Ayrıca bu robot üzerinde kullanılan sensörler, daha karmaşık görevlerin gerçekleştirilebilmesi için gerekli veriyi bilişsel modellere iletmekte yetersiz kalacağından, daha sonraki çalışmalar için Darwin-OP insansı robotu tercih edilmiştir. Darwin-OP insansı robot, üzerinde taşıdığı mini-bilgisayar ile hareketli bir bilgisayar özelliği taşımaktadır. Ubuntu işletim sistemi aracılığı ile de daha farklı modellerin çalıştırılmasına imkan sağlamakta, ayrıca gömülü bulunan kamerasıyla çevreyi algılamayı da başarabilmektedir.

Tez çalışmasında ele alınan bilişsel süreçlere ilişkin hesaplamalı modeller, sinirbilim konusunda yapılan çalışmalar ile belirlenen beyindeki ilgili yapıların özellikleri ve bağlantıları gözönüne alınarak geliştirilmiştir. Bu hesaplamalı modellerin etkinliğini, özellikle ortam ile etkileşimini test etmek için robotlar üzerinde "ödül öngörülü uyaran" görevi kullanılmıştır. Bu görevde robotlar, öncelikle ortamdaki uyaranları, onaylanan bir hareket ile eşleştirmeyi, yine ortamdan alacakları ödül ile öğrenebilmiştir. Sonrasında, hesaplamalı modeldeki kimi bağlantıları ödül öngörüsü ile pekiştirip, bastırılarak eşleştirmeyi öğrendikleri, bu uyaran-hareket çiftine ait gösterimi, değiştirebildikleri de gösterilmiştir. Böylece, aynı uyaranı farklı bir hareket ile eşleştirmeyi, yine ödüle bağlı olarak tekrardan öğrenebileceği gösterilmiştir. Bu uyaran- hareket eşleştirme görevi sırasında kullanılan uyaranlar renk kartlarıdır. Kırmızı, sarı ve mavi renk kartları robotun hareket uzayında tanımlı olarak bulunan üç hareketle eşleştirilmiştir. Kullanılan robotların özellikleri, ve gerekli yazılımsal donanımlar ile ele alınan görev Bölüm 2'de tanıtılmıştır.

Tanımlanan ödül öngörülü uyaran görevinde robot, kamerasını kullanarak algıladığı renk uyaranına karşılık bir hareket seçmektedir, bu harekete karşılık bir ödül alırsa, sonrasında bu renk uyaranını gördüğünde istenilen hareketi seçmeyi pekiştirmektedir. Görevde tanımlanan hareket seçme işlemi temelde beynin bazal ganglia, talamus ve korteks (BTK) bölümlerinin etkinliği ile ilişkilendirilmektedir. Bütüncül olarak

baktığımızda beyindeki birçok devre hareket seçimine etki ederken, temelde bu üç bölümün ele alınması, bilişsel süreçlere ait modellerin kullanışlılığını ve gerçek süreçlere ait verilerin elde edilmesinde yararlı olabileceğini göstermek açısından yeterlidir.

Bundan dolayı, beynin BTK parçaları ele alınan görevi gerçeklemek için farklı seviyelerde modellenmiştir. Öncelikle BTK devresi olarak doğrusal olmayan dinamik sistemler bakış açısıyla, bir grup sinir hücresinin etkinliğini modellemede yararlanılan yığın modeli yaklaşımı ile modellenmiştir. Bu model, beynin bölümlerinin birbiriyle ilişkisini tanımlayan fark denklemlerinin çözülmesiyle hareket seçimini gerçekleştirmektedir.

BTK yığın modelinin biyolojik gerçekçiliği olmasına rağmen, nöron seviyesinde bir modelin sağlayacağı biyolojik öğrenme kurallarının etkisini inceleme şansını sağlamaz. Yığın modeli, beyin yapılarının davranışlarını bir nöron popülasyonunun davranışlarının ortalaması olacak şekilde fark denklemlerine indirger. Bu tez çalışmasında nöron seviyesindeki modelleme de ele alınmış ve BTK yapısı modellenirken korteks yapısı nokta nöronlar ile modellenmiştir. Tüm modeli daha gerçekçi olan nokta nöron modelleri ile gerçeklemek istememize rağmen, sadece korteksin nokta nöronlar ile gerçeklenmesi, bu tezin kapsamında nokta nöron modelinin robotlar üzerinde somutlaştırılabileceğini göstermek açısından yeterlidir. Hesaplamalı modeli, oluştururken ele alınan bu farklı yaklaşımların yanı sıra öğrenme süreci için de yapılara bağlı olarak farklı yaklaşımlar ele alınmıştır.

Robotun öğrenmesini sağlamak için biyolojik gerçekçiliğe sahip hareket seçme devresinin kullanılması yanında yine biyolojik olarak anlama sahip bir pekiştirmeli öğrenme yöntemi olan zamansal farklarla öğrenme yöntemi kullanılmıştır. Bu yöntem ile hareket seçimi ve hareket seçiminin ardından ortamdan gelen ödül kullanılarak hareket seçimine ait modelin parametreleri makine öğrenmesi yaklaşımı ile güncellenmektedir. Böylelikle uyarana karşı seçilen hareket de değiştirilmiş olur. Biyolojik olarak anlama sahip olmasına karşın kullanılan yöntemin makine öğrenmesi metodu olmasından dolayı bu yöntemi nokta nöron modeli ile elde edilen devrelere uygulamak zordur. O yüzden vuru zamanına bağlı plastisite (STDP) yöntemi nokta nöron modelleri ile kullanılmak üzere gözönüne alınmıştır. Bu yöntem de zamansal farklarla öğrenme yöntemi gibi ödülü kullanmakta, ancak nokta nöronlarla modellenmiş yapılar arasındaki bağlantıları ödüle ve nöronların vuru zamanlarına bağlı olarak değiştirmektedir. Yığın modelleri üzerindeki öğrenmeden farklı olarak bu modelde hücre seviyesinde öğrenme de ele alınmaktadır. Dolayısıya, yığın modeli ile sadece zamansal fark metoduna dayalı pekiştirmeli öğrenme kullanılırken, korteksin nokta hücre modeli ile gerçekleştirldiği durumda, STDP ile zamansal fark metodları öğrenme için kullanılmıştır.

Tezde, ilk olarak bazal ganglia, talamus ve korteksten oluşan yığın modelindeki parametreler zamansal fark öğrenme yöntemi kullanılarak güncellenmiş ve ödül öngörülü uyaran görevi gerçekleştirilmiştir. Yığın modeli içinde yer alan parametreler, çevreden gelen uyarana karşılık modelin seçeceği hareketin belirlenmesinde etkindir. Ele alınan modelde öğrenme için güncellenen parametreler, Wc ve Wr, sırasıyla çevreye ilişkin oluşan algıyı ve modeldeki dopamin seviyesini ifade eder. Ele alınan beyin yapıları arasındaki bağlantıları etkiliyen parametrelerin (Wc ve Wr) ve gelen uyaranın ne kadar kuvvetli aktarıldığının, öğrenme üzerindeki etkisi incelenmiştir. Böylelikle tezdeki amaçlardan biri olan hesaplamalı modeller aracılığı ile ele alınan bilişsel süreçte rol alan nöral yapıların etkinliğinin incelenmesine ilişkin sonuçlar elde edilmiştir. Yığın modeli ile elde edilen sonuçlar Bölüm 3'de verilmiştir.

Yukarıda da değinildiği gibi, özellikle ödül öngörülü öğrenme için makine öğrenmesine ilişkin bir yapı olan zamansal fark yöntemi yerine biyolojik olarak daha gerçekçi bir öğrenme kuralı ile ele alınan bilişsel süreci modellemek amacıyla, gerçeğe uygunluğu daha fazla olan nokta nöron modelleri ele alınmıştır. Nokta nöron modelleri her ne kadar kablo denklemleri ile ifade edilen ve sinir hücrelerinin morfolojik özelliklerini de içeren modellere göre basit olsa da temel yapı olarak sinir hücresinin özelliklerini barındırması ve hesaplama yükünün daha karmaşık modellere göre oldukça az olmasından dolayı gerçekçilik-performans ölçütünde önemli bir avantaja sahiptir. Bundan dolayı ele alınan hareket kararına ilişkin hesaplamalı modeli daha gerçekçi bir yapıya taşımak için nokta nöron modeli kullanılmış ve BTK yığın modelinde bulunan korteks nokta nöron modeli ile değiştirilmiştir. Böylelikle iki modlu bir hesaplamalı model ile hareket seçimi görevi gerçekleştirilmiştir. Bunu sağlamak için iki farklı boyuttaki modelin çalışma aralıkları birbirine uygun hale getirilmiştir. Korteksteki belirli zaman aralığında eşik değerini geçerek, vuru üreten nöronların sayısının ortalaması alınıp $0-1$ arasında bir değere ölçeklenerek yığın modeli denklemlerine bir terim olarak eklenmiştir. Aynı şekilde yığın modelindeki değişkenlerin değerleri ölçeklenerek nöron girişlerine akım olarak eklenmiştir. Böylelikle iki modelin eş zamanlı çalışması sağlanmıştır. Yığın modeli ile nokta nöron modellerinin birlikte Darwin-OP insansı robot üzerinde gerçeklenmesi sırasında NEST nöral simülasyon kütüphanesi kullanılmıştır. Darwin-OP'un motor komutlarının bulunduğu ve C++ ile kodlanmış kısım ile Python ortamı üzerinde çalışan NEST kütüphanesinin birlikte çalışması sağlanmıştır. Böylelikle gerçekçi bir hesaplamalı modele ait gerçek zamanlı çalışma, Darwin-OP üzerinde test edilmiştir. Bu sonuçlar Bölüm 4'de verilmiştir.

Biyolojik gerçekçilik için ilk adım olarak BTK modelinde kortekse ilişkin model, yığın modeli yerine nokta hücre modeli ile değiştirilse de öğrenme için zamansal fark yöntemi yerine, vuru üreten sinir ağları için kullanılan STDP öğrenme yöntemi Bölüm 5'de ele alınmıştır. STDP, birbirine sinapslarla bağlı iki nöronun vuru üretme sürelerine bakarak aralarındaki sinapsları ödülü de kullanarak kuvvetlendiren ya da zayıflatan bir öğrenme yöntemidir. Bu yöntem de NEST kütüphanesi kullanılarak vuru üreten sinir ağları modeli ile oluşturulmuş korteks ile birleştirilmiştir. Böylelikle ödül öngörülü uyaran görevi basit ama gerçekçi modellere sahip olarak gerçeklenmiştir. Basit bir modelin hareket seçimi için kullanılmasının sebebi, zamansal faktörlerin önemli olduğu STDP'nin hareket seçim devresi ile birlikte kullanılabileceğinin gösterilmesi ve özelliklerinin araştırılmasının işlem yükü ve zaman açısından daha avantajlı olmasındandır.

Bu çalışmalar sonucunda, hesaplamalı modeller farklı seviyelerde gerçeklenerek hem bu modellerin gerçek sonuçlar ile ilişkisi gözlenmiş, hem de bu modellerin gerçek zamanlı görevler için robot üzerinde gerçeklenmesi sağlanmış oldu. Yığın modeli ile yapılan çalışma sonucunda, modele verilen uyaranların sırasının modelin öğrenme başarısında ve süresinde etkili olduğu belirlendi. Bununla birlikte, uyaranın alt birimlere aktarılma kuvveti de öğrenmenin gerçekleşmesi için önemli bir yere sahip olduğu gözlemlendi. Alt birimlere uyaran bilgisinin çok fazla aktarılması, istenilen hareketlerin seçilmesini engellerken, bu aktarımın az olması da öğrenmenin hiç sağlanamamasına sebep olmaktadır. Ayrıca, dopamin seviyesini belirten parametre değerinin çok yüksek olması modelin aynı anda birden fazla seçim yapmasına sebep olmaktadır. Yine talamus aktivitesinin fazla olması aynı anda birden fazla seçimin yapılmasına sebep olduğundan öğrenme gerçekleşmemektedir. Bu gibi sonuçların yığın modeli üzerinden elde edilmesi, hesaplamalı modellerden anlamlı bilgiler çıkarılabileceğini göstermektedir. Yığın modeli ile davranışsal açıklamalar yapmak daha mümkün olmasına karşın, vuru üreten sinir ağları ile elde edilmiş modellerin de detaylı bir şekilde incelenmesinin, beynin çalışmasına ait bu gibi sonuçların hızlı bir şekilde elde edilebilmesine olanak sağlayabileceği gösterilmiştir.

Böylelikle özellikle hayvan modelleri ile yapılan deneysel çalışmalarla çok deneme yapmak yerine, bu çalışmalara hızlı bir şekilde yön verecek sonuçların hesaplamalı modeller ile elde edilebileceğine ilişkin bir sonuç bu çalışma ile verilmiştir. Bu sonuçların yanında vuru üreten sinir ağları modeli ve STDP öğrenme yöntemleri de insansı robot, Darwin-OP, üzerinde gerçeklenmiştir. Darwin-OP üzerinde ödül öngörülü uyaran görevinin gerçeklenmesi de daha karmaşık görevlerin de gerçeklenerek beynin çalışmasına ait sonuçlar elde edilebileceğini göstermektedir.

İlerleyen çalışmalarda, tezde kullanılan hareket seçimine ait hesaplamalı modellere, serebellum ve hipokampüs gibi motor kontrol, navigasyon ve algı ile ilgili beyin bölümleri eklenerek daha karmaşık görevler tasarlanabilir. Beynin bu bölümleri için gerekli olan yeni bilgiler için ise, kullanılan robota farklı sensörler eklenerek ortamdan gerekli farklı bilgi sağlanabilir. Böylelikle robotun çakıllı, kumlu, kaygan vb. ortamlarda da hareketi ve bilişsel görevleri gerçeklemesi sağlanabilir.

1. INTRODUCTION

Embodiment is a concept that found its place not only in psychology and philosophy but also in robotics and artificial intelligence. As a word it means a tangible or visible form of an idea, quality or feeling, but in this study we will focus on its meaning in cognitive science. From perspective of embodied, embedded cognition (EEC), brain body and world, all are important factors in explaining how an intelligent behavior emerges. Neurorobotics is a mean to create a testing environment for EEC [1, 2]. In this thesis, the leading idea is to establish an example of implementing a model for a cognitive process based on neuroscience studies. Thus, with embedding a computational neuroscience model in a humonoid robot, and rendering learning of a cognitive task with the interaction of robot and environment, a step will be taken toward embodiment.

Embodiment involves the interaction with body and environment. From this perspective the most significant ability for animals and humans is movement which is provided by one of the most studied circuit in the brain, motor circuit. In [3, 4], it is stated that basal ganglia, which is a neural structure in the midbrain, has recurrent connections to cortex and thalamus and the loop generated by basal ganglia, thalamus and cortex (BTC) are highly associated to motor circuit in the brain. The process of generating movement is handled by a channel-like separated circuits in BTC loop in relation to brainstem and cerebellum in general [3–5]. Although the brainstem and cerebellum networks are linked to providing the required patterns for movement and feedback modulation, BTC itself is associated to the embodiment process which is explained as both selection and initiation of an action [5]. In [6], BTC loop is also referred to action selection circuit in the brain because of its movement initiation and termination abilities.

In [7], a sequence learning task is dealt from a working memory aspect and basal ganglia is considered as responsible for action selection with related cortex and thalamus parts. In that study, basal ganglia makes decisions by using its "Go" and " No go" pathways emerging from striatum. These two main pathways arising from striatal D1 and D2 type dopamine receptors take care of direct or indirect pathways which corresponds to the "Go" and "No go" pathways. These direct and indirect pathways implement the initiation and the termination of a movement or selecting one movement instead of other movements in brain and this process is provided by dopamine network in the striatum [3]. Besides considering reward based learning for action selection as a cognitive process of BTC and trying to have a model for motor actions, it is expected here that such modelling would be versatile for developing new diagnosing and treatment procedures.

It is reported in [3, 8] that some of the disorders in the dopamine network of basal ganglia show up as Parkinson's disease (PD), Tourette syndrome (TS) and Huntington's disease (HD) which are related to the abnormal voluntary movements. The symptoms of PD show itself as inability of initiating a voluntary movement, involuntary slowness and shaking while the TS and HD are associated to the uncontrollable movements apart from their mental disfunctions as reported in [3, 8]. Since these cognitive disorders are associated to the disfunctioning in the BTC loop and dopamine network, it is important to understand the process of initiation and termination of movement in motor circuit.

Since BTC circuit is highly related to movement, in this thesis a substructure is prepared for a test environment for complex task which could include mobility. To realize such a basic test environment BTC loop is modeled in different scales for the implementation on a robot and some results are obtained for the process of BTC loop. However, beyond the embodiment, the relation between learning and BTC circuit is also discussed. In [9], Schultz et al. states that reward based learning process in the brain explained as having better predictions of future rewards and this is associated to the dopaminergic activity in the basal ganglia. And this dopamine activity is used to shape the future predicions in other words experiences. So, they claim that basal ganglia and its dopaminergic network plays an important role in reinforcement learning (RL) in which the connections between structures are modulated by reward in response to sensory cues and the defects in the network result in behavioral disorders such as addiction and obesity [9]. Therefore, the temporal difference learning (TDL) is utilized as a RL method to provide learning of BTC circuit considering biological plausibility.

In the recent studies of basal ganglia, there are many spiking neural networks (SNN) simulated for action selection tasks. In [10], the authors constructed the cortico-thalamic pathway including striatum, subthalamic nucleus (Stn), globus pallidus internal, globus pallidus external which are substructures of basal ganglia. In this work, the stimulus arises directly from cortex and all the neuron groups consist of Adaptive Exponential Integrate and Fire neurons which are defined with four differential equations. Only one stimulus takes place in this work to be associated to two different actions. In addition, they consider probabilistic rewarding and the agent may get reward as long as desired results are obtained even if the selection was not correct. Learning is applied to the connectinos between cortex and striatum, subthalamic nucleus and thalamus, respectively. The modulation is realized with spike timing dependent plasticity (STDP) which utilizes reward signal. Chersi et al. [11] investigate the relation between goal-directed and habit driven systems with a stimulus-action association task in which a monkey tries to learn to turn on the desired lamb in a simulated environment according to flashing lights. The considered network is constructed by SNN structure which contains basal ganglia in relation with sensory and motor cortices and thalamus. Their approach benefits separated groups of neurons and each of them represents a channel in a neural structure. All neuron populations are modeled by leaky integrate and firing neurons and the learning occurs between sensory structures, striatum, Stn in addition to the prefrontal cortex and motor circuit. The modulation method is STDP with reward modulation. So, they try to mimic the behaviour of action selection in some of the related brain parts in simulation.

Besides SNN models, the dynamical system models are also utilized to investigate effects of neural parameters on the action selection. In [12], the authors discuss the dopamine effect on an action selection mechanism that takes part in basal ganglia in a simulated robot environment. They utilize the same BTC circuit that contains channels for each action and model the network with using dynamical system models. The neuromodulation of dopamine is modeled with differential equations as well and the learning is realized with the basis of Hebbian learning. Another dynamical system model of BTC circuit is utilized in the study of Prescott et al. [6] to realize an action selection task inspired by navigation of a rat in an unfamiliar environment. The BTC circuit is modeled with difference equations of neural structures and implemented to a mobile robot in a hard-coded way without a plasticity or learning. By using the model on a mobile robot they investigated the relevance of model to the findings from experimental results. In [13], Sengor et al. modeled the BTC loop to simulate a goal-directed behaviour with difference equations. The main difference of this work is to implement learning into the dynamical system from the reinforcement learning approach and this work also is a basis for the model used throughout this thesis. The reward modulated BTC circuit is extended for a simulated robot task in [14, 15] with the investigation of parameter space effect on action selection by bifurcation analysis.

Based on the previous works summarized above, there are numerous studies on basal ganglia loops not only in neuroscience but also in computational neuroscience. Our aim and approach could be considered as trying to build a connection between neuroscience and mathematical modeling and engineering and gathering results where both parties could benefit. So, different aspects of BG circuit is considered. Since BG circuit plays an important role in embodiment besides providing decision making and learning, a cognitive task has to be defined to test the feasibility of the computational models. In addition, it is important to implement computational models on robots from the embodiment aspect which may provide more information about the underlying cognitive processes of complex tasks. So, to take a step towards using the BTC loop for the complex tasks in real environment, a simple temporal sequence task is defined to implement the BTC loop in different levels on a humanoid robot. In the considered task, robots learn to associate and reassociate a sensory stimuli to desired actions with respect to the given reward.The properties of the utilized robot and the softwares with the handled task are given in Chapter 2.

At first, the BTC loop is modeled as a mass model with difference equations and the model parameters are updated by using temporal difference learning (TDL) that utilizes reward coming from environment. The mass model selects an action according to the sensory stimulus. Updated parameters of the mass model represent the perception of environment and dopamine level. To investigate how the model is meaningful to derive information, the effects of values of parameters that lie between neural structures and strength of stimuli on learning is examined. In addition, the effectiveness of neural structures are examined by tampering the connections within the BTC loop. The mass model is implemented on a robot and results are given in Chapter 3.

Even the learning method, TDL, is biologically meaningful, it is still a machine learning method. To model the learning process in a more realistic way, point neurons are considered. Point neurons are not as realistic as morphologically modeled neurons, but they are efficient for computation and still contain the basic properties of neurons. That's why the cortex part of mass model is changed with point neuron based model in Chapter 4. The considered mass model and the point neuron based cortex work together to decide on an action during the temporal sequence task. The implementation of the considered mix model to humanoid robot, Darwin-OP, and the results are explained in Chapter 4.

Changing cortex part of the mass model with point neurons, which are SNN model, is the first step for the biologically plausible model. Another step is changing the TDL method for learning. So, STDP model is utilized instead of TDL. STDP modulates the strength of synapses considering spike timing of pre and post neurons and the reward coming from environment. The SNN model of cortex and STDP are utilized to realize temporal sequence task and implemented on Darwin-OP. The results of implementation are given in Chapter 5 with the explanation of the structure.

In this thesis, the computational models are realized in different levels and not only the relation of model results with real experiments but also the implementation of these models on robots in real time are provided. Therefore, different from other SNN based implementations of action selection, this model is implemented on a real robot and the model run online in real time. So, this work opens a door for the investigation of fully point neuron based realistic models from the embodiment aspect in order to obtain information about the process of brain functioning and disfunctioning.

2. TASK AND ENVIRONMENT

As pointed out in the introduction, basal ganglia circuits have role not only in motor actions and learning but also role in embodiment [16]. So, here we will first define a task, where a cognitive process, uniting association of sensory information with motor actions, which is important for proprioception and thus for embodiment and learning together: temporal sequence task. Then, the properties of the robot used and the simulation environment will be described. Thus, a computational model of a cognitive task is implemented on a humanoid robot with learning methods to show the applicability of building associations between sensory stimuli and desired actions in real time. So, a step toward realizing a neurorobot which is capable of realizing intelligent behavior with a dynamic model of neural structures is taken. Though the task is simple and all the features of the humanoid robot are not utilized, still the results are intriguing for embodiment, too.

Through this study, a temporal sequence task is considered that is performed by macaque monkeys, where it is expected to match a stimulus with an appropriate movement [17]. The same task is realized with different computational models in the following sections to investigate the cognitive behaviour on the humanoid robot.

In the considered task, the robot is expected to associate the presented colors to the desired predefined actions. As shown in Figure 2.1 on the right side, there are three colors that are yellow, blue and red to be associated to the predefined actions which are head movement, leaning and hand movement, respectively. The green color is used to indicate reward given to the robot if its decision is the desired one corresponding to the color shown.

During the task, the robot learns matching three different stimuli, which are different colours, with three different predefined movements. The robot differentiates colours using its camera, and the colour recognized is the input of the computational model, where action selection is done. The action selection process is depicted on Figure 2.2.

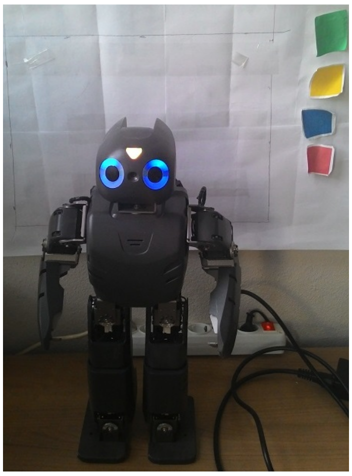

Figure 2.1: The robot used in the study is a humanoid robot platform called Darwin-Op. The humanoid robot is expected to associate the presented colors to the desired predefined actions. The three colors to be associated to the actions are red, yellow and blue. The green color is used to indicate reward given when the action choice is the desired one.

When, a colour is presented to the robot, it is expected to select an action in the first place. With the computational model implemented, the robot tries to decide on an action. If it cannot decide, the action is realized based on random selection. Green colour is shown to represent reward, to indicate that the action realized is a proper one. Once the robot is rewarded due to a right choice, an expectation error arises, which updates the parameters of computational model in charge of action selection. Once the update is completed by learning rules, the colour is shown again and the correct action is rewarded each time until robot learns to match the appropriate movement with the colour.

In addition to this, robot is also expected to rearrange the previously associated sensory input-action pairs when the rewarded pairs are changed. In this way, robot can manage to associate the sensory stimulus to a new desired action by reward and change its previous behaviour. It will be shown that, the implementation of the computational model on humonoid robot also shows this adaptation capacity of model to the changing environment.

Even though different robot platforms has been used in similar works [6], [15], in this study humanoid robot platform called Darwin-Op which is shown in Figure 2.1 is preferred. This humanoid robot is chosen because of its high capacity for interaction with the environment. In the previous studies a computational model that is built as dynamical system model for action selection is utilized ([18], [19]). In these studies,

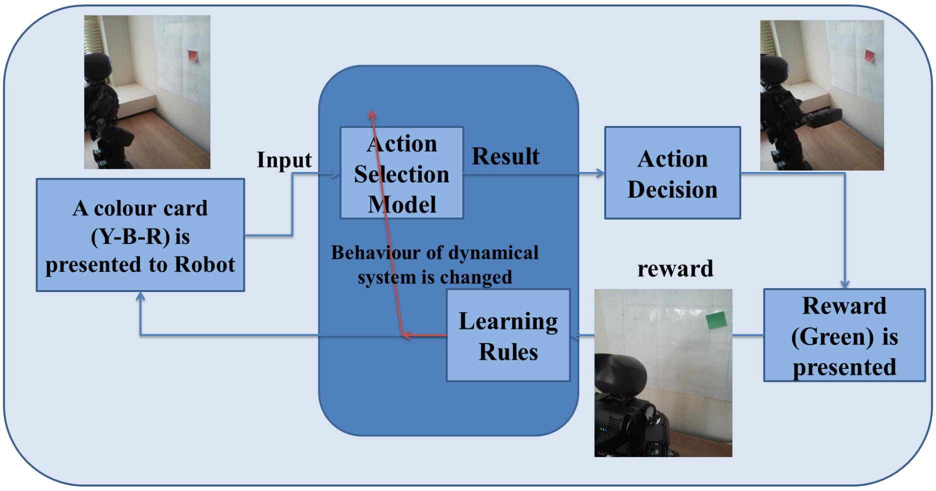

Figure 2.2: The process of task in real environment. At first, the robot is presented a stimulus and the computational model that is responsible of action selection, makes an action decision. According to desirability of this decision, a reward is given to the robot. This reward is evaluated by learning rules and changes the behaviour of computational model with updating its parameters. And the repeated process makes the robot learn how to associate a stimulus to an action.

Bioloid robot platform is utilized for the implementation. Since the computational load of model was lower than the ones in this thesis, Bioloid realized the task well in real time. However, Bioloid has little storage capacity and its computational power is not enough to implement an operation system on it. Because of these reasons, it is not possible to simulate the computational models based on point neurons in real time. In addition, variety of implementable sensors on Bioloid is limited to recognize environment. Considering all these limitations, the Darwin-OP humanoid robot is selected for the investigations of the computational models that take part in through this thesis. So, instead of making effort on the manipulation of robot or the construction of vision, focus on the biologically realistic models are preferred since Darwin-OP handles those with its pre-defined scripts. Darwin-OP has a built-in camera among its eyes and the color information is obtained by using it. It has 1.6 GHz Atom CPU and 1GB RAM inside and all the calculations are realized on the robot and in real-time. The robot consists of 20 servo motors. Darwin-Op uses Ubuntu as an operation system and the codes to control its motors and the dynamical system model of BTC loop are programmed in C++. It is also possible to use different software tools and libraries for the computation of cognitive models since Darwin-OP has Ubuntu operation system. Therefore, a structure is prepared for further experiments that may use more realistic vision data or locomotion in the environment by utilizing Darwin-OP.

In the oncoming parts of the thesis SNNs are utilized with the dynamical system model. Apart from the dynamical system model of BTC loop, SNN model is simulated on NEST simulator. That's why Python is also utilized on Darwin-OP. NEST is a point neuron based simulator and it is designed to investigate the dynamics of neuron populations instead of considering the exact morphologies [28].

Even the implementation of the model and learning method changes with SNN structure, the cognitive process remains same for the task.

3. COLOR ASSOCIATION TASK USING BTC MASS MODEL AND TDL

The neural structures taking part in the temporal sequence task introduced in Chapter 2, compose of subcortical structures as striatum, globus pallidus externa/interna and subthalamic nucleus together with frontal cortex and thalamus, and all these structures are considered in the computational model of BTC loop proposed for action selection. The model of BTC that will be implemented on the processor of Darwin-OP to realize the temporal sequence task, is a mass model, where the activity of a population of neurons are represented by nonlinear difference equations. Since to realize the temporal sequence task, not only action selection but building association between sensory inputs and their representations and reward based learning is needed all these processes will be modeled as updating the parameters of the dynamical system corresponding to BTC loop, through reinforcement learning using TDL method [15]. The process of task in real time is realized on Darwin-OP as explained in Chapter 2.

In the following sections, first the equations governing the BTC model and the learning rule will be introduced, then the results of the experiments carried for the temporal sequence task will be given.

3.1 BTC Mass Model and TDL

The BTC mass model consists of the difference equations (3.1,3.2, 3.3) that construct the dynamical system. Each equation represents the averaged behaviour of related neuron population in discrete time. The equations given here are the modified versions of the equations in [13, 15].

BTC model consists of relations between substructures of Basal ganglia, thalamus and cortex all of which are parts of a rat brain [4]. The scheme of these relations is given in Figure 3.1 which shows the excitatory (arrowed lines) and inhibitory (pointed lines) connections between substructures of Basal ganglia (BG), cortex and thalamus. The connections between these substructures indicate positive or negative contribution to

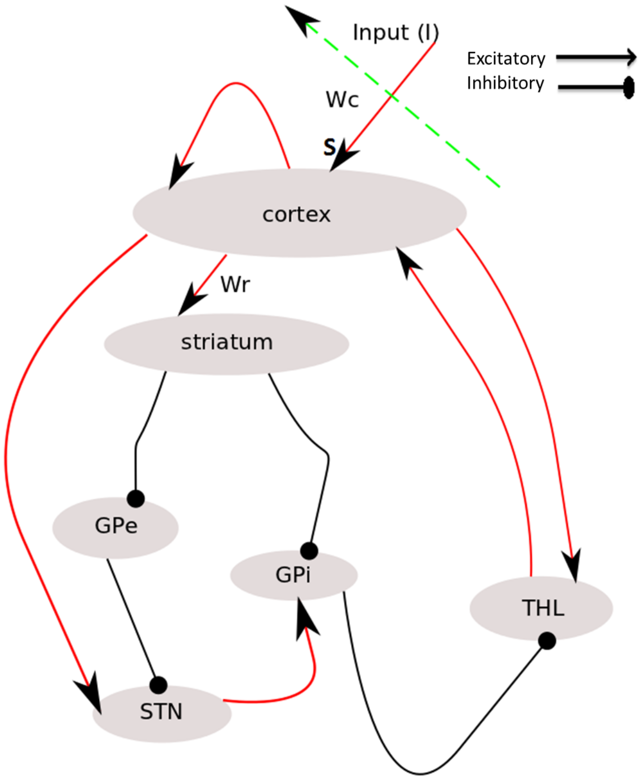

Figure 3.1: Block diagram of Basal ganglia (BG) circuit: This diagram shows the excitatory (arrowed lines) and inhibitory (pointed lines) connections between substructures of BG , cortex and thalamus.

the values of the parameters that are given in Equations 3.1, 3.2. The substructures of BG considered in the model are striatum (Str), globus pallidus external (GPe), subthalamic nucleus (Stn) and globus pallidus internal (Gpi).

$$S(k) = W_c I(k) \tag{3.1}$$

$$\begin{aligned} Ctx(k+1) &= f(\lambda Ctx(k) + Thl(k) + S(k)) \\ Str(k+1) &= Wrf(Ctx(k)) \\ GPe(k+1) &= f(-Str(k)) \\ Stn(k+1) &= f(Ctx(k) - GPe(k)) \\ GPi(k+1) &= f(Stn(k) - Str(k)) \\ Thl(k+1) &= f(Ctx(k) - GPi(k)) \end{aligned} \tag{3.2}$$

Action selection part of the study is realized by an iterative calculation of the Equations 3.1, 3.2. Equation 3.1 indicates the linear relation between information coming from the environment (I) and inputs of the cognitive model (S). So, Equation 3.1 models the association between stimuli and its representation in the cortex. Equations 3.2 models the interrelation between BG substructures, cortex and thalamus ([14]).

There are three different actions to be selected through the task and three sensory inputs all of which are explained in Chapter 2. So, all the variables related to the brain areas are in vector form and their dimensions are $3x1$. Each element of the vectors stands for a channel on the related brain structure. If the task considered had more sensory inputs, then the dimension of the vector would be more than three.

k is the discrete time variable for all of the equations. S indicates the representation evoked due to the sensory inputs (I). This relation between the sensory inputs and the representation in the cortex is built up as a linear transformation by W_c matrix that is a $3x3$ matrix. W_c is the adaptive connections between the stimuli and cortex and indicates their significance in the environmental context. W_r is the other adaptive connection weight between Ctx and Str and represents the effect of dopamine on action selection. Its dimension is $3x1$ since there are not intrachannel connections in the model except W_c. The modification of W_c matrix and W_r vector changes the behavior of dynamical system and the selected action as a result.

S activates Ctx which is the input structure of BTC model. After the activation of BTC model, the decision making process begins and the result of action selection is determined by the values of Ctx at the end of the cycle. Once the variables of BTC model converge to an equilibrium point [13], the Ctx values determine the action selected. λ coefficient denotes the recurrent behavior within the cortex.

The function $f(.)$ is a tangent hyperbolic function, and it is used to model the mean activity of the neuron populations. The $f(.)$ function is given by Equation 3.3.

$$f(x) = 0.5\,(tanh\,(3\,(x-0.45))+1) \tag{3.3}$$

Modulation of the connections between inputs and model with the internal connections are provided according to the TDL. TDL is a reinforcement learning method ([20]) that is claimed to be related to reward based learning in basal ganglia [9]. TDL modulates the connections by evaluating the expectation error. When an agent decides on an action, it has an expectation on the result of that action. The action changes the environment and the difference between the reward obtained and the expectations due to the new state of the environment of the agent arises an expectation error. In the considered task, the expectation of agent on reward and the given reward determine this error.

The modulation of the connections is provided by the Equations 3.4 to 3.8. In these equations, k indicates discrete time and all k dependent parameters are in vector or matrix form except V and δ_c which are scalar variables. η and μ are constants and their values are both 0.9. r stands for the reward information coming from environment. Its value is 1 when there is reward and 0 otherwise. In Equation 3.4, V indicates the value assigned to the given inputs. This value information is kept in W_v which has one weight value for each input, so it is a $3x1$ dimension vector.

$$V(k) = W_v(k) I(k) \tag{3.4}$$

$$\delta_c(k+1) = r + r\mu V(k-1) - V(k) \tag{3.5}$$

$$W_v(k+1) = W_v(k) + \eta\,\delta_c(k) I(k) \tag{3.6}$$

$$W_c(k+1) = W_c(k) + \eta\,\delta_c(k) Ctx(k) I(k) \tag{3.7}$$

$$W_r(k+1) = W_r(k) + \eta\,\delta_c(k) Ctx(k) Str(k) \tag{3.8}$$

The expectation error (δ_c) is calculated according to the given reward and the difference between previous and current values that is denoted in Equation 3.5. This expectation error modulates the value weights of inputs W_v according to Equation 3.6. So, the weights of values are modulated using the input information and the expectation error when there exists sensory information. Also, W_c matrix and W_r vector, which indicate the weights of connections on the action selection model, are modulated using the expectation error due to the reward obtained as a result of action.

Thus, whenever there is a difference between the expectation and the actual result, the connections between the neural structures, W_c and W_r, are updated proportional to the relation between cortex and inputs for W_c and between cortex and striatum for W_r. W_r determines the projection of information to the basal ganglia. After cortex begins to select the desired actions in sequence, the W_r connections increase with respect to the expectation error and this increases the projection of information to the basal ganglia which effects the learning in long term. In this study, W_r connections have a base value that loosely corresponds to the base level of dopamine in the model.

3.2 Investigation of Parameters: Experiments and Results

The BTC model consists of difference equations that constructs a dynamical system. It is well-known that the change in the parameters of a nonlinear dynamical system gives rise to change in the behavior of the system and bifurcation analysis is a tool to investigate this phenomena. As, in [21] and [15] this analysis is already carried out, here a number of computer aided experiments will be carried out to see the effect of parameters on the system behavior more explicitly.

As change in the system's behavior corresponds to learning in this context, the experiments focus on learning. Thus, learning is dependent to the initial values of the parameters from the dynamical systems aspect. As an experiment, the first investigation that will be presented is the effect of initial values on learning. Then, it is shown that after learning is accomplished, the BTC model can accurately select actions for related inputs as a second experiment. In this case, the BTC model on the robot can realize action selection with learnt parameters and there is no need to update the parameters again. As a last experiment, the success of this model on explaining some biological connections between the Basal ganglia, thalamus and cortex is investigated. These experiments besides grasping the learning experience, also intends to comprehend the role of dynamical environment on the behavior. Each experiment is first realized as a computer simulation and then on the robot environment.

3.2.1 Dynamically changing environment with different initials

Dynamically changing environment can be expressed in two ways. The first way is that the initial perception of the decision making circuit (BTC model) on the environment (W_c) can be different for different experiments. This means that the W_c parameter determines the perception of the environment ([22]) and if we initially select W_c different for different trials, then we would be able to model the initial perception of the environment by the BTC model. The second way is that the values of sensory inputs can be different for the same stimulus or the sequence of the stimulus can vary from experiment to experiment. The first way is investigated on simulation and on the robot by randomly changing W_c for each trial. The second way is investigated on the robot since the sequence of the inputs will vary during experiment. The experiments are realized on a different robot environment, which is Bioloid that is explained in Chapter 2, instead of Darwin-Op, but since only the sequence of inputs are considered during the experiments and the input values given to the model are the discretizated ones, the results would be same for Darwin-OP.

The initial values of W_c parameter may cause BTC model not to be able to learn, since the initial values have effect on the convergence of dynamical system model of BTC [21]. That's why the effect of choosing different initial values for W_c is firstly investigated on simulation. The results are given in Table 3.1. In this experiment, there are two success rate for two subexperiments. In both of the subexperiments, only one channel input is given a higher value than the others at a time. At the first one, the higher channel inputs are selected as 0.9 while the lower ones are selected as 0.1. At the second subexperiment, the higher channel inputs are selected as 1 while the lowers are selected as 0. As it is expected, the success rates of both are below 100%. When only W_c parameter's initial values are selected randomly and the other parameters are same for each trial, the success rate of the first subexperiment is 83% and of the second subexperiment is 76% considering 10000 number of experiments for each case. So, massive number of experiments are carried out to have statistically robust results. This means that the BTC model can accomplish 83 of a hundred learning trials with random W_c initials and the same value of the other parameters when the inputs are given to the model in regular order. The regular order means that the inputs are given in a repetitive sequence and not in a random order.

The percentage differences between the systems that is stimulated with $0-1$ and $0.1-0.9$ values are assumed to be caused by the zero value of the low level input, which blocks the model to use the information coming from low level inputs during the learning process due to the multiplication by zero during updating the parameters. On the other hand, the percentages of successful trials are higher when the inputs are given to the model in a random order (Table 3.1). The number of mean steps in the table indicate the number of states, after which the model selects the desired action for the given input and the expectation error decreases to below 0.01 for the successful trials. Not only the number of steps for whole process to be terminated but also the number of steps that are needed to accomplish the successful learning are higher in the random order case. This means that the dynamical system model converges to its

Table 3.1: Results of learning with different W_c initials.

High	Low	Success	# of $Step_{Mean}$	Order of Inputs
0.9	0.1	% 83	248.5	In Regular Order
1	0	% 76	215.9	In Regular Order
0.9	0.1	% 95	344.1	In Random Order
1	0	% 82	285.2	In Random Order

desired fixed points (actions) for the given inputs in longer time but more strongly with random order inputs than regular order inputs.

Some of the initial values of W_c for successful trials are given in Table 3.2. In this table, it is seen that the number of steps before learning are different for different initials. The initial values of W_c are updated with TDL rules by using reward given from experimenter. So, W_c evolves to a value that BTC model makes decisions accurately. The evolved values of W_c for the fourth initials, which are given in Table 3.2, are presented in Equation 3.9. Considering W_{c_f}, the diagonals of the matrix is higher than the other which indicates that the model may associate the inputs to the actions in certain conditions. The certain conditions are related to W_r, which stands for dopamine effect, range of inputs and connections of neural structures that will be investigated in Subsection 3.2.3.

Table 3.2: Learning duration for random W_c initials.

$Param.$	1^{st}	2^{nd}	3^{rd}	4^{th}
w_c^{11}	1.82	-1.2	1.11	-0.5
w_c^{12}	0.54	0.22	0.36	-0.3
w_c^{13}	1.38	0.86	0.39	-0.6
w_c^{21}	-0.9	-0.1	-0.5	0.27
w_c^{22}	0.9	0.23	-0.5	1.28
w_c^{23}	-0.3	0.41	0.05	-1.0
w_c^{31}	1.36	-0.2	-0.1	-0.4
w_c^{32}	-0.2	-0.5	0.13	-1.2
w_c^{33}	0.95	-1.0	-1.7	-0.8
# of steps	683	236	246	198

$$W_{c_i} = \begin{bmatrix} 1.82 & 0.54 & 1.38 \\ -0.9 & 0.90 & -0.3 \\ 1.36 & -0.2 & 0.95 \end{bmatrix} W_{c_f} = \begin{bmatrix} 13.5 & -1.8 & -2.8 \\ -9.0 & 15.1 & -9.5 \\ -0.35 & -10.2 & 16.1 \end{bmatrix} \quad \textbf{(3.9)}$$

In this study, the selected action is expected to follow the high valued inputs which means that when the input of first channel is high then the first action is the desired one and so on. In Figure 3.2, the outcomes of learning process are presented. The initials and evolved values for the Figure 3.2 are the ones that are given in 3.9. The upper figure of Figure 3.2 shows the selected action for the given input and the lower one shows the expectation error through learning. Red line seen in upper figure shows the channel number of higher input and blue line indicates the selected action for given input. At the beginning, input doesn't match with the selected action and expectation error is high. At the end of experiment, model manages to select the right action for the given input and expectation error is close to zero. In the middle of learning process (Figure 3.2), the BTC model selects right action for a given input. While the expectation error is decreasing to zero, a wrong choice of the action causes higher expectation error than before (370^{th} step in Figure 3.2). These wrong decisions make the model learn to select desired actions more precisely at the end of the experiment while the expectation error decreases to zero. So, the selected actions (blue lines) and inputs (red lines) matches on the figure. This is the desired process of BTC model on the learning task.

The same learning experiment is repeated with inputs in random order. The initial W_c values are chosen randomly and the other parameters are left same with the previous experiments, results of which are given in Table 3.1. One of the successful learning experiment with random ordered inputs can be seen in Figure 3.3. Considering this experiment, the order of inputs affect the expectation error, which determines whether learning is achieved or not. It is clearly seen from Table 3.1 that learning trials are more successful when the inputs given to the model are in random order.

3.2.2 Results after learning is accomplished

After learning is accomplished and the BTC model successfully associates given input to the desired action, the expectation is that the BTC model can successfully associate inputs to the desired actions by using the same parameters without updating them. The results after learning is accomplished is given in Figure 3.4 and 3.5. The W_c parameter values are initially taken as W_{c_f} and are given in (3.9). The learning process

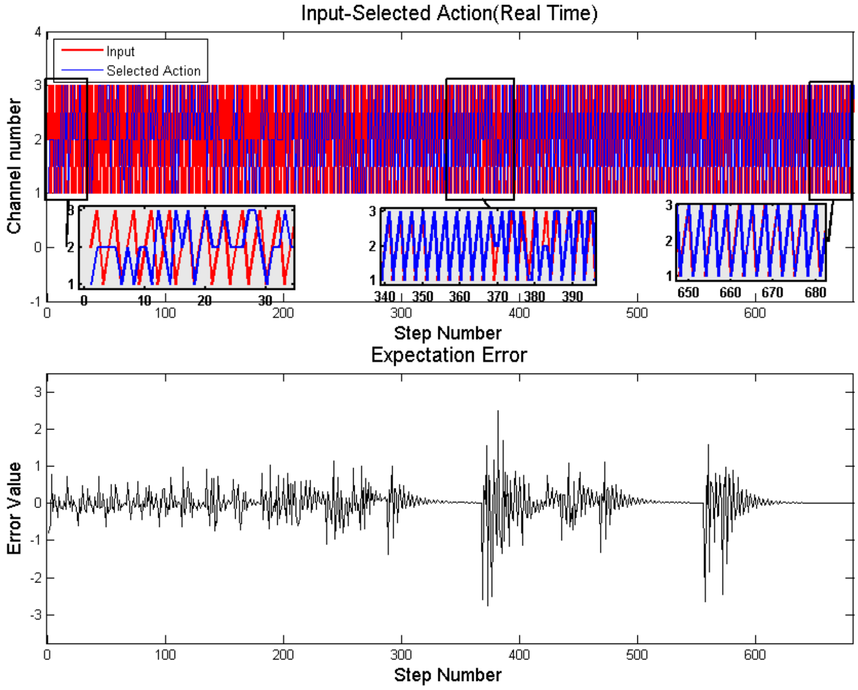

Figure 3.2: The upper figure shows the selected action for the given input and the lower one shows expectation error through learning. Red line seen in upper figure shows the channel number of higher input and blue line indicates the selected action for given input. At the beginning input doesn't match with the selected action and expectation error is high. At the end of experiment, the model manages to select the right action for the given input and expectation error is close to zero.

realized to obtain these parameters can be seen in Figure 3.2. The order of inputs through learning is regular. In Figure 3.4, these parameters are taken and the model is tested with inputs in regular order. In Figure 3.5, the inputs are in random order, and the BTC model successfully selects desired actions for given inputs. There is no difference between random and regular order inputs after learning is accomplished. However, if the maximum value of input for a channel, input value of which is the highest, decreases the model may not select the desired action even it has learnt to select before. The effect of maximum value of inputs is explained in Subsection 3.2.3.

3.2.3 Tampering the connections between neural structures

One of the aims of this model is to explain the effect of some connections between cortex, basal ganglia and thalamus on action selection task. To explain these relations, we have tampered the connections between neural structures given in Figure 3.1. Besides the connections between neural structures given with the Equations (3.2), the values of model parameters have an important role on the learning accomplished by

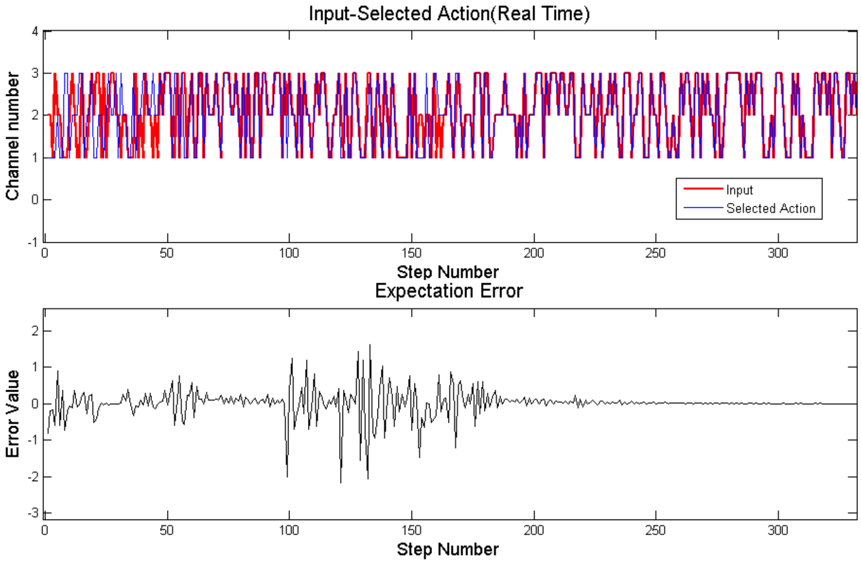

Figure 3.3: The upper figure shows the selected action to a given input with random order. The lower one shows expectation error through learning. Red line seen in upper figure shows the channel number of higher input and blue line indicates the selected action for given input. At the beginning input doesn't match with the selected action and expectation error is high. At the end of experiment, model manages to select the right action for the given input and expectation error is close to zero. With random ordered inputs, model learns better than learning with regular ordered inputs.

the model. Two of these parameters are $W_{r_{base}}$ and the maximum value of inputs (S_{max}) presented to the BTC model. W_r is a parameter between cortex and striatum, and it represents the role of dopamine level on action selection in the model. By changing its value, how the dopamine level effects the action selection process can be observed. The value of W_r is between 0 and 1. $W_{r_{base}}$ is the base level of W_r and the value of W_r is kept bigger than this base level through learning process. S_{max} is the maximum value of the inputs, and it is taken to be between 0 and 1. The value of S_{max} limits the information coming from environment to cortex for action selection. As seen in Table 3.3, these two parameters have effect on success percentages. The learning process is repeated 10000 times for each value pair and the number of successful trials are obtained. The number of successful learning trials are obtained with regular order of inputs through learning and random initial W_c values for each of 10000 experiments. The BTC model is the most successful when $W_{r_{base}}$ is 0 and S_{max} is 0.5. S_{max} is the level of information sent to cortex and learning rates decrease when S_{max} is increased for the same $W_{r_{base}}$ considering the BTC model. The lowest learning rates are obtained when S_{max} is 1 (excluding $S_{max} = 0.4$ situation) and this means that the zombie situation

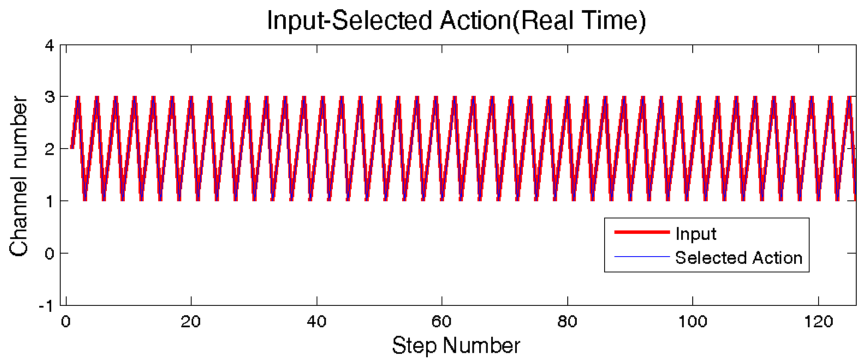

Figure 3.4: After learning, the BTC model can successfully associate inputs to the desired actions by using the same parameters without updating them. The W_c parameter values are initially taken as W_{c_f} given in (3.9). In this experiment, the order of inputs through learning is regular. Figure shows the selected actions for the given inputs.

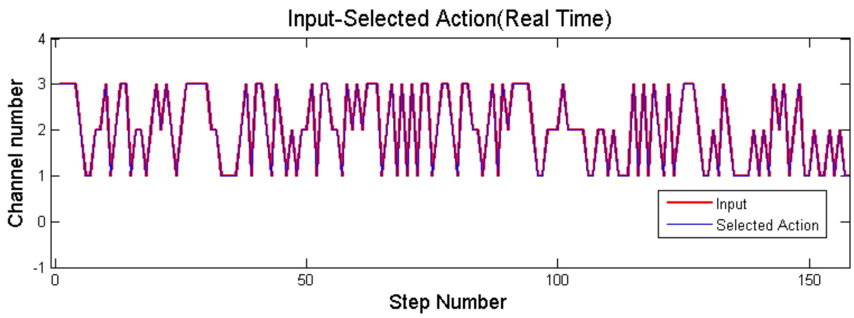

Figure 3.5: After learning, the BTC model can successfully associate inputs to the desired actions by using the same parameters without updating them. The W_c parameter values are initially taken as W_{c_f} given in (3.9). In this experiment, the order of inputs through learning is random. Figure shows the selected actions to given inputs.

($S_{max} = 1$) [23] is not much successful as the others to change its selection for different types of input. As it can be followed from Figure 3.6, the model stucks between two actions at the end of this experiment and cannot select the third one. Besides, the expectation error is low through the experiment and close to zero at the end even though it doesn't get reward. The number of successful learning trials also decrease while increasing $W_{r_{base}}$. Increasing base level of W_r makes the BTC model to select more than one action at a time. Thus, the increase affects values of W_c ,which evaluates sensory inputs, and the inputs are perceived wrongly because of high W_r level. In addition to this, high W_r level increases the activity of thalamus on cortex, and as a result cortex is urged to select more than one channel at a time.

In Table 3.3, it is given that when S_{max} is 0.4, the model is not successful on learning to select right action. This is because the input is not big enough to make cortex select

Table 3.3: Success percentage (%) for different S_{max} and $W_{r_{base}}$ values.

		$W_{r_{base}}$							
	Success Rate	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7
S_{max}	0.4	0	0	0	0	79.6	27.5	7	0
	0.5	95.7	94.3	90.6	84.7	71.9	22	9	0
	0.6	84.6	83.4	82.6	78.2	72	23.9	5	0
	0.8	76.3	77.4	74.5	74.1	68	18.1	5.1	0
	1.0	68.4	71.7	70	71	64.8	18.7	4.2	0

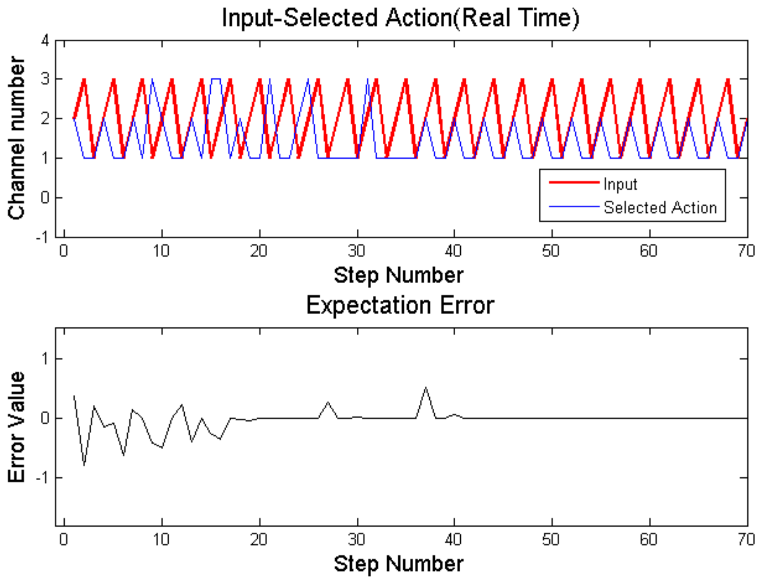

Figure 3.6: Zombie situation ($S_{max} = 1$). The learning experiment is realized with random initial W_c and $W_{r_{base}}$ is 0. The model stucks between two actions at the end of learning experiment and cannot select the third one. The expectation error is low through experiment and close to zero at the end despite it doesn't get reward.

an action. However, when $W_{r_{base}}$ is 0.4, the number successful learning trials are higher than before since W_r increases the activity of thalamus and this situation compensates the low values of inputs.

Giving inputs, that is only one of the three inputs has the high value, in different order changes the learning process. Inputs in random order prevents the model to stuck on an undesired input-action pairs by changing the expectation error. Because of this, the number of successful learning trials for experiments with random order inputs are higher than with regular order inputs as stated in Table 3.1. This can be seen in Figure 3.7. In the Figure 3.7, colors indicate the number of successful learning trials for each

S_{max}-$W_{r_{base}}$ pair. 10000 learning trials are simulated with random initial W_c for each of S_{max}-$W_{r_{base}}$ pairs. The colors show the number of successful learning trials in 10000 trials. The figure on the left hand side of Figure 3.7 shows the number of successful learning trials with using inputs in regular order and the right-hand figure shows the number of successful learning trials with using inputs in random order. Comparing two, the number of successful learning trials of random order inputs are higher than regular order inputs. However, the order of inputs doesn't effect the dark blue areas, which indicates unsuccessful learning.

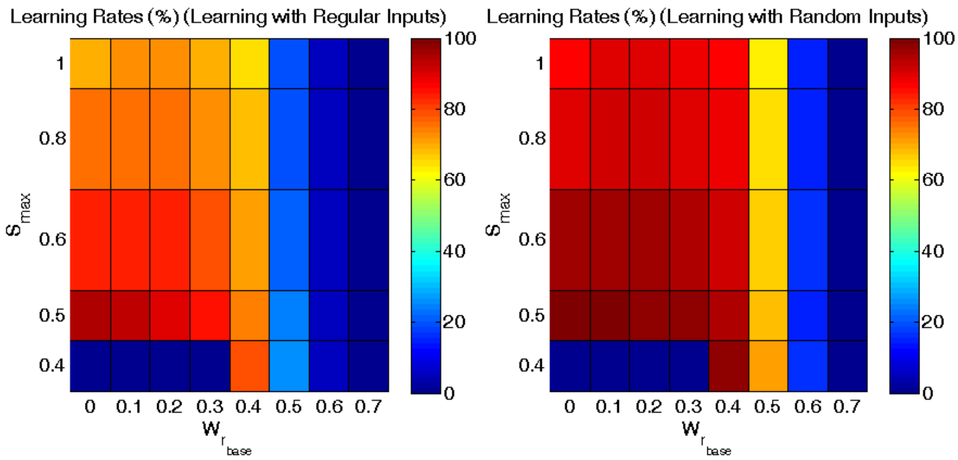

Figure 3.7: The number of successful learning trials with respect to S_{max} and $W_{r_{base}}$. The left-hand figure shows the number of successful learning trials with using inputs in regular order and the right-hand figure shows the learning rates with using inputs in random order. Giving inputs to the model in random order increases the number of successful learning trials, but it doesn't affect the zero areas.

While the order of inputs has an effect on the number of successful learning trials, it does not make learning possible for the parameter values, for which the model cannot learn to select the desired action. On the other hand, changing connections between neural structures in the BTC model can change the behaviour for different parameter values. The values of connections are between 0 and 1. In Figure 3.8, three of excitatory connections seen in Figure 3.1 (Thl to Ctx, Stn to GPi and Ctx to Stn connections) are reduced by half and the number of successful learning trials are investigated. By reducing Thl-Ctx connection it is seen that there is not much difference when $W_{r_{base}}$ is lower than 0.5. However, the number of successful learning trials are reasonably higher when $W_{r_{base}}$ is increased to 0.5 or higher. When W_r is high, the model tries to select more than one actions at a time and thalamus activity highly affects cortex. So, reducing the connection from thalamus to cortex also reduces this thalamus activity and makes cortex available to select one action at a time. This situation increases learning rates for higher values of $W_{r_{base}}$. As it is stated before, when S_{max} and $W_{r_{base}}$ are 0.4, low inputs are compensated with the high level of W_r and because of this, thalamus activity is depressed. However, decreased thalamus activity cannot compensate this situation and makes the learning rate zero for these parameter values. Another set of experiments is realized by reducing the excitatory connection between Stn and GPi. As given in lower left hand side figure of Figure 3.8 learning rates become zero when $W_{r_{base}}$ is greater than 0.1. In the other situations learning rates are significantly low with respect to the reference situation (upper left figure). When the connection between Ctx and Stn is decreased by half, the figure in the lower right hand side of Figure 3.8 is obtained. The number of successful learning trials are even lower than the other situations. In the last two situations, the excitatory connections in the BTC model through Stn are disrupted separately which means that the information coming from input and Ctx cannot transferred to GPi. Thus, GPi cannot inhibit Thl enough. When W_r is low (lower than 0.2 for this experiment) some channels are selected instead of the desired channels because of the disinhibition of Thl for most of the experiments. When W_r is higher than 0.2, all channels are selected at a time because of the high activity of Thl. That's why the number of successful learning trials are low when the path through GPi is disrupted. According to the results of the BTC model high Thl activity on cortex decreases possibility of selecting the desired action as more than one is selected. This shows a kind of hiperactivity, which impairs the action selection process.

3.3 Results on Simulation and on Humanoid Robot

Humanoid Robot, Darwin-OP, is expected to learn how to associate the given stimuli to the predefined desired actions and how to rearrange the associations to accomplish the task that is explained in Chapter 2. The model is tested on a MATLAB simulation before realizing on Darwin-OP.

At first, the model is trained to associate the first stimulus (red color) to the first action, the second stimulus (yellow color) to the second action and the third stimulus (blue color) to the third action. The result can be seen in Figure 3.9. The inputs are given to the model in order. The upper figure shows the relation between inputs and selected

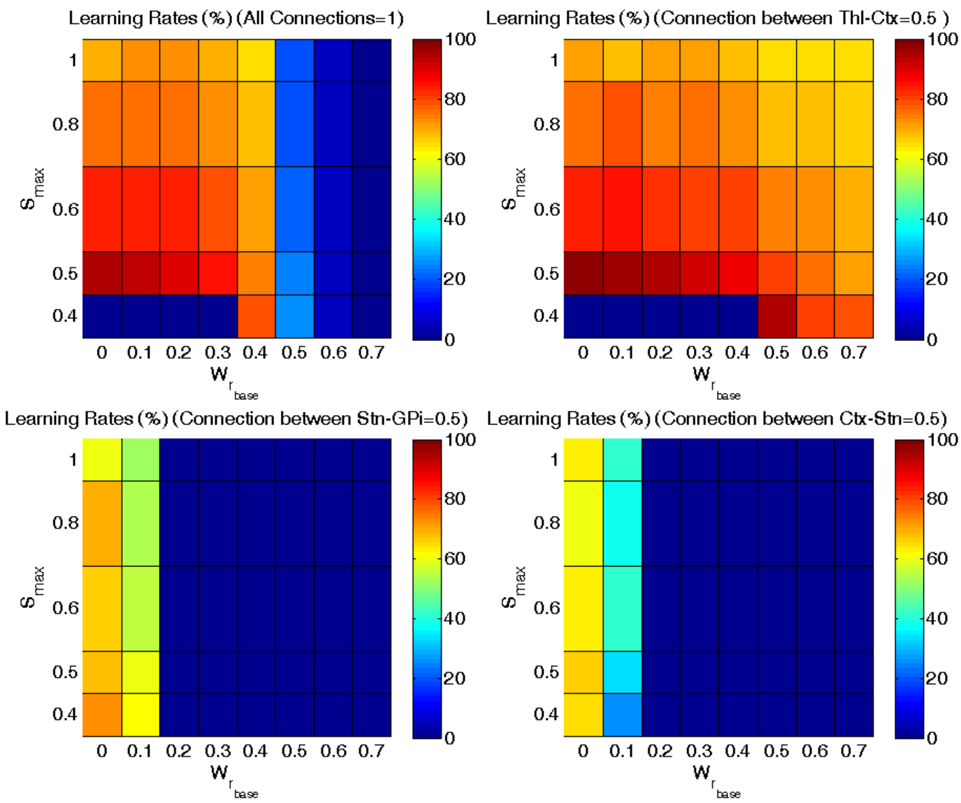

Figure 3.8: The number of successful learning trials with using inputs in regular order for different connections between neural structures. Thalamus (Thl) to cortex (Ctx), subthalamic nucleus (Stn) to globus pallidus internal (GPi) and Ctx to Stn connections are reduced by half. When Thl activity is decreased on Ctx, the number of successful learning trials increase (upper right). When excitatory Stn connections are disrupted, this decreases the number of successful learning trials since Thl activity on Ctx is increased by disrupting (lower figures).

actions and the lower figure shows the expectation errors and reward. When the value of reward line is high, that means the model gets reward. At the first encounter with reward, the expectation error rises as expected and through the experiment decreases exponentially to zero with using the reward and manage to find right W_c and W_r parameter values. At the end of the experiment W_c and W_r reach values seen in 3.10 and 3.11.

$$W_c = \begin{bmatrix} 8.68 & -8.11 & -0.56 \\ 0.11 & 23.14 & -2.55 \\ 0.11 & -7.5 & 9.14 \end{bmatrix} \tag{3.10}$$

$$W_{r1} = \begin{bmatrix} 0.8275 \\ 0.9986 \\ 1 \end{bmatrix} W_{r2} = \begin{bmatrix} 1 \\ 0.6466 \\ 1 \end{bmatrix} W_{r3} = \begin{bmatrix} 0.9108 \\ 0.9098 \\ -0.0556 \end{bmatrix} \tag{3.11}$$

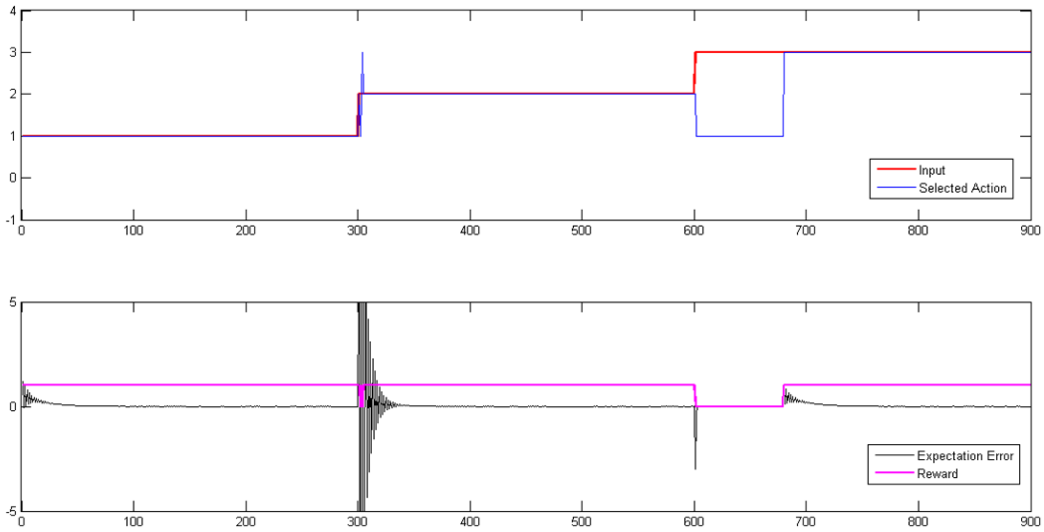

Figure 3.9: Results of associating colors to the actions in MATLAB environment. In this figure, first stimulus is associated to the first action, second stimulus to second action and third stimulus to third action as seen in the upper sketch. The lower one shows the expectation error and reward during the task. 1 in the upper figure indicates the first stimulus or action, 2 indicates the second stimulus or action and 3 indicates the third stimulus or action with the related line colors.

After the first associations, they are changed by using reward and the first stimulus is associated to the third action and the third stimulus is associated to the first action as given in Figure 3.10. The expectation errors are higher at the time of first encounter of unexpected reward. Even the association between the second stimulus and second action is left same, an expectation error occurs since the values related to this association are also updated through the task. The reason of this is that when the first stimulus is given to the model it tries to select the second action, which is wrong, and the values related to the second action are changed during the update. The new parameters at the end of changing the associations are given in Equations 3.12 and 3.13.

$$W_c = \begin{bmatrix} -3.06 & -8.11 & 6.38 \\ -1.19 & 1.88 & -11.6 \\ 0.3 & -7.51 & -18.6 \end{bmatrix} \tag{3.12}$$

$$W_{r1} = \begin{bmatrix} 1 \\ 1 \\ 0.5497 \end{bmatrix} W_{r2} = \begin{bmatrix} 1 \\ 0.5167 \\ 1 \end{bmatrix} W_{r3} = \begin{bmatrix} 0.5498 \\ 1 \\ 1 \end{bmatrix} \tag{3.13}$$

In the real-time experiments Darwin-OP is presented the stimuli as explained in the Chapter 2. The results are seen in 3.11. The upper figure shows the input and action

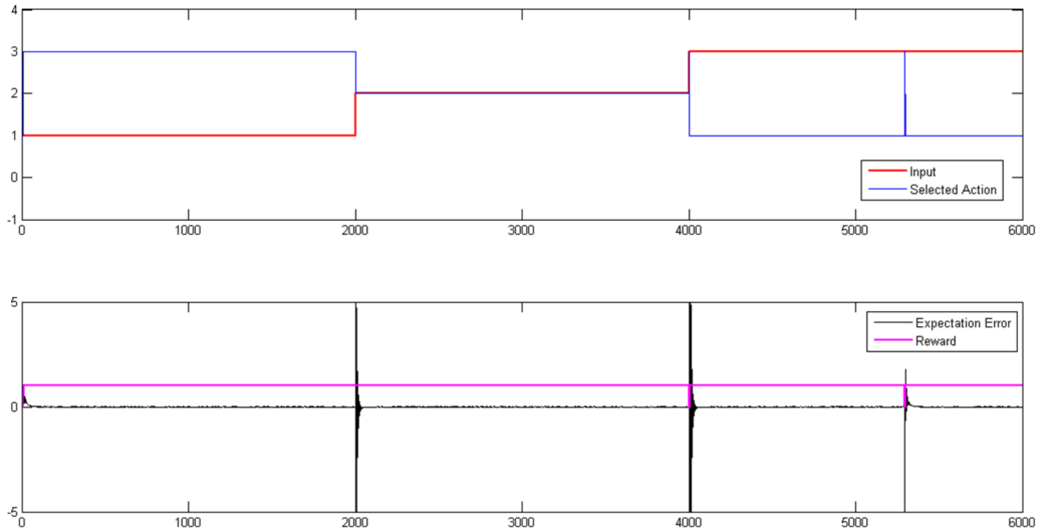

Figure 3.10: Results of rearranging the previously associated stimulus-action pairs in MATLAB environment. The upper sketch shows the relation between the inputs and actions and the lower one shows the expectation error and reward. 1 in the upper figure indicates the first stimulus or action, 2 indicates the second stimulus or action and 3 indicates the third stimulus or action with the related line colors.

relation as explained for the MATLAB results. The red line indicates the presented color's number and the blue line indicates the number of selected action. The lower figure shows the change of expectation error during task. At the beginning of the experiment the first color presented to the robot. It manages to learn associating the first color to the first action after several trials and getting reward for the correctly selected actions. Then the second and third colors are presented in sequence. At the end of the building of first association (at the $35^{th} step$), the value of W_c matrix is given in 3.14 as $W_{c_{first}}$. After the first part, the first color is presented again, but the reward is given selecting the second action instead of the first one. After robot manages to rearrange the first association, third and second colors are presented in sequence as seen in Figure 3.11 (after $35^{th} step$). The third color is newly associated to the first action and the second color is newly associated to the third action. However, since the expectation error is stuck at zero, it takes more trials to rearrange the association of second color. At the end of experiment the new W_c matrix get the value ($W_{c_{second}}$) in 3.14.

$$W_{c_{first}} = \begin{bmatrix} 2.2 & 0.05 & 0.18 \\ 0.09 & 1.97 & 0.03 \\ -0.77 & 0.18 & 1.96 \end{bmatrix} W_{c_{second}} = \begin{bmatrix} 0.15 & 0.05 & 1.91 \\ 1.87 & 0.17 & 0.03 \\ 0.14 & 1.9 & 0.17 \end{bmatrix} \quad \textbf{(3.14)}$$

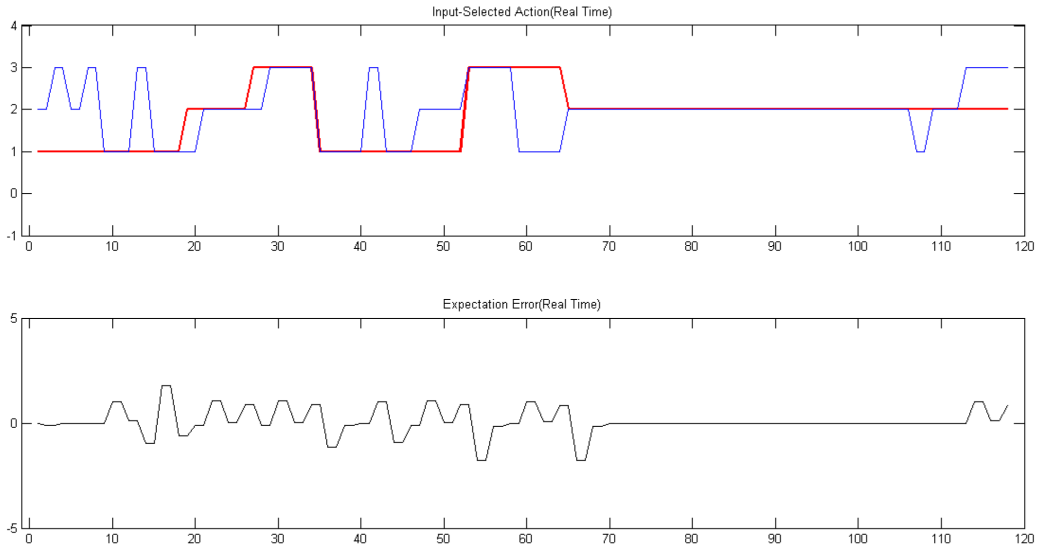

Figure 3.11: Results of associating the color inputs to the actions. The red line in upper figure indicates the number of color presented to the model. The blue line indicates the selected action's number at the time of presentation of the color. The change of expectation error through the task is shown in the lower figure. The first association finishes at the 35^{th} step and rearranging the associated pairs begins after that time.

Humanoid robot is trained to associate a stimulus to an action successfully in real-time. It can rearrange the association it has set up to rebuild a connection between the stimulus and a new action. Even these experiments show the compatibility of the model and robot in real-time and the ability to show the relations between some of the structures of brain on action selection, the neuron based models have to be investigated to reach a deeper understanding of the action selection process and to close the gap between the morphology based models and the behavioral models. That's why a step to neuron based models has taken in the following chapter.

4. THE COLOR ASSOCIATION TASK USING SNN AND MASS MODELS

In the previous section,it is shown that the model imposed is able to show different behavior and even behavioral deficiencies due to the change in parameters. Nevertheless, the model is still far from being able to give a better understanding of action selection, especially due to update rules based on reinforcement learning. So, in order to have a biologically realistic model of reward based learning, but still to have a realization that can be implemented on robot, as a first step, a small population of point neurons considered instead of mass model for cortex.

Though, the cortex is composed of point neurons instead of mass model, the reward based learning, still is accomplished by temporal difference learning as in Chapter 3. So, building up the association between sensory inputs and the desired actions are built up, in a similar way. In this chapter, first the differences made on the model will be explained, then how point neuron model is implemented on humanoid robot will be given. The experiments carried out in this case will be explained and the chapter will be concluded with discussion on the results.

4.1 Neurocomputational Model

In this part of study, the BTC model considered in [13, 15] is expanded using spiking neural network model of cortex which is given in Figure 4.1. Like the model in Chapter 3, sensory information which reaches to cortex is transferred to basal ganglia and thalamus through cortex and processed there to decide on an action. The main difference in the neurocomputational model is, the neural structures are modeled in two different scales: point neurons and mass model So the model is in a way mixed mode model. The sensory information transfer is realized by three different pathways: Direct pathway through striatum (Str) and globus pallidus internal (GPi), indirect pathway through Str, globus pallidus external (GPe) and subthalamic nucleus (Stn) and hyperdirect pathway through Stn and GPi [24, 25].

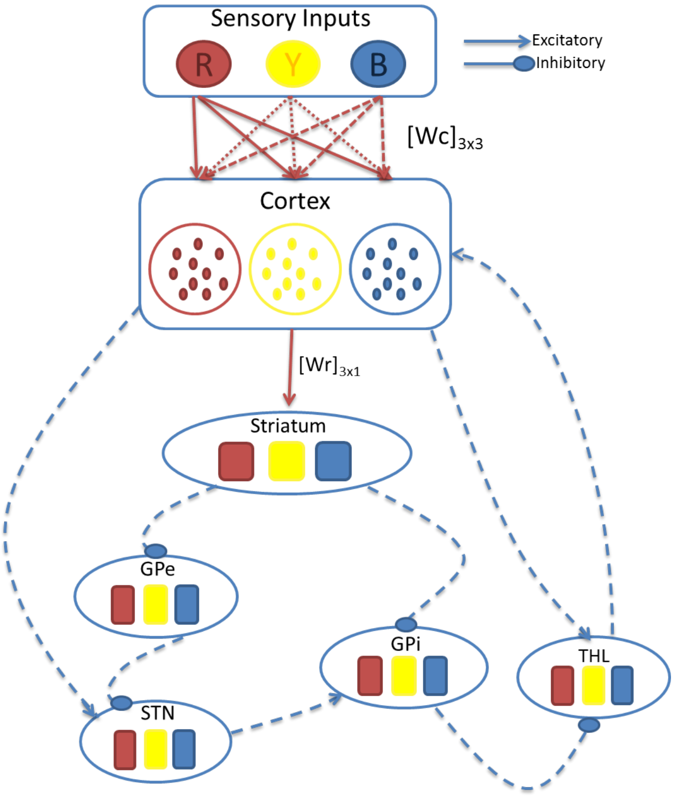

Figure 4.1: Basal ganglia-Thalamus-Cortex (BTC) action selection model. Model is structured with the connections between cortex, basal ganglia substructures and thalamus. Cortex part of the model consists of point neurons while the other structures are modeled as mass models. Basal ganglia part consists of striatum (Str), globus pallidus external (GPe), globus pallidus internal (GPi) and subthalamic nucleus (Stn).

Since in the task considered, there are three sensory information, cortex has three separated neuron populations, which are named as channels. In Figure 4.2, these three neuron populations/channels are indicated by three different colors. In addition, each channel has two neuron groups: excitatory and inhibitory neurons, which are denoted by upper groups and lower groups in Figure 4.2, respectively. Excitatory neurons make connections only within the channel. They have random connections to themselves and to the inhibitory neurons of the channel. However, inhibitory neurons are connected to the excitatory neurons of each channel. So, when a channel is promoted by a specific sensory stimulus, it inhibits the other channels by its inhibitory neuron group. In this way, the information that is transferred through W_c is strengthened by these inhibitory connections and the winner-take-all structure that is established by W_c connections is highlighted [26].

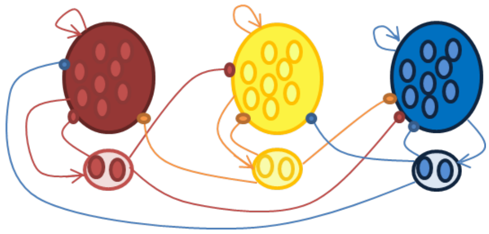

Figure 4.2: Spiking neural network model of cortex. There are three channels in the cortex model each for a sensory stimulus and each channel consists of 80 regular spiking and 20 fast spiking Izhikevich point neurons [27] connectivity of which are 10%. Regular spiking neurons are excitatory (upper neuron groups of each channel) and they have only connections inside its channel. Fast spiking neurons are inhibitory (lower neuron groups of each channel) and they have interchannel connections. So, the connections between channels are provided by inhibitory neurons of each channel.

Excitatory neurons and inhibitory neurons are modeled as regular spiking and fast spiking point neurons, respectively. The equations used to model the point neurons, which are Izhikevich neurons, can be found in [27]. Each of the neurons consists of two differential equations and these equations are solved in time. When the state variable v of neuron equations exceeds a threshold, the neuron fires which is also called as spike activity, and the state of the neuron is reset to the initial value. However, NEST model of Izhikevich neurons are utilized instead of solving these differential equations explicitly during the task.

There are 80 regular spiking and 20 fast spiking neurons in each channel of cortex. All connections are realized with 10% random connectivity. So, each regular spiking neuron makes eight random connections in the excitatory neuron group of the channel and another eight random connections to fast spiking neurons of the same channel. And each fast spiking neuron makes two random connections to each of three excitatory neuron groups of all channels.

The basal ganglia and thalamus neurons are modeled as mass models and each of them also has three channels. Cortex, thalamus and each element of basal ganglia are connected as shown in Figure 4.1. Since cortex is modeled as SNN and other structures are as mass model, the cortex output is transferred to the basal ganglia after a process. Outputs of cortex are the number of spikes that the related channel has. Cortex part of the model is simulated 200 *ms* for one cycle that is explained with Figure 2.2. So, the cortex part is simulated 10 *ms* with NEST and this is repeated in 20 steps to complete one cycle. At the end of each step the spike counts of related channel is scaled into $0-1$ interval and this value is sent to striatum. For the basal ganglia part the Equations 3.2 are solved iteratively except the cortex equation and the result of thalamus which is between 0 and 1 is sent to the cortex for the next step. The detailed explanation of the solving BTC equations takes part in [18]. The value sent from thalamus to cortex is added as a synaptic current to the equations of cortex neurons that be found in [27] as I. In this way, 20 steps are carried out to complete one cycle of action selection. And then, according to the action result taken from cortex, the reward comes from environment to modulate the connections that are given in Figure 4.1.

In Figures 4.1 and 4.2, the excitatory connections are shown as regular arrows and inhibitory connections are shown as point-headed arrows. All the connections are static except the ones between sensory stimuli and cortex and between cortex and striatum, which are indicated as W_c and W_r respectively. These dynamic connections are modified to build up the association between sensory stimuli and actions. In this way, the connections and neural structures shown in Figure 4.1 compose a dynamical system model of BTC circuit.

Now, dynamical connections between sensory stimuli and cortex, i.e.,W_c and between cortex and striatum,i.e., W_r will be explained in more detail for the modified model. Though the cortex part added as SNN to the model in Chapter 3, the meanings of W_c and W_r are same with the previous model. Each sensory stimulus, which corresponds to red, yellow and blue colors are denoted by R, Y and B letters in Figure 4.1, has connections to excitatory neurons of all three channels. Each sensory stimulus connects to all excitatory neurons of the three different channels similarly. The value of promoted input, the input of presented color, is 0.9 while the values of other inputs are 0.1. In this way, all inputs take part in the learning process but with different importance. In addition to the sensory stimuli, the excitatory neurons in cortex have noisy inputs with Poisson of 45 Hz. So, there are nine dynamic connections from sensory stimuli to the three different channels of cortex, which builds up $3x3$ matrix W_c. Due to this connection structure, before association is built, the sensory stimuli are homogeneously connected to each channel though there are different channels denoting three different colors.

The other dynamic connection is between cortex and striatum. Each channel of cortex projects onto the same channel of striatum. The projection to striatum is proportional to firing rates of the excitatory neurons of channels in cortex. Therefore, there are three connections through the channels of cortex and striatum and these connections are indicated as W_r which is denoted by a $3x1$ matrix. These dynamic connections between sensory stimuli and cortex and between cortex and striatum are modulated with expectation error of TDL as explained in Chapter 3.

4.2 Implementation on Humanoid Robot

The humanoid robot platform, Darwin-OP, is utilized to realize the task which consists of associating colors to the desired actions and rearranging associations as explained in Chapter 2. Ever so the process to realize the task is same with the explained in the previous chapters, the communication scheme and calculation of selected action is realized in a different way.

Model is coded in two parts on the humanoid robot which are the module responsible for action selection and the module responsible for parameter adaptation to accomplish learning. The action selection model is coded in Python environment using NEST simulator for the spiking neural network part [28]. In addition to this, getting sensory input and actuation part is coded in C++ with learning included. The communication scheme of the communication between two environment can be seen in Figure 4.3. At first, the humanoid robot gets sensory inputs with its camera and sends this information with the weights of connections, W_c and W_r, to the simulator part. In the simulator part, Python coded part, the decision is calculated using sensory inputs and connection information. Then the calculated cortex and striatum information is carried to the C++ coded part for getting reward and updating the connection. The two environments wait for the results of the other on real-time process of task. In this way, structurally different two models are merged to run in the same environment.

4.3 Experiments and Results

In this study, the humanoid robot is expected to select the desired actions when specific colors are presented. Thus, it is expected to learn to associate the sensory stimulus to an action by evaluating reward and also to rearrange the previously learnt pair for

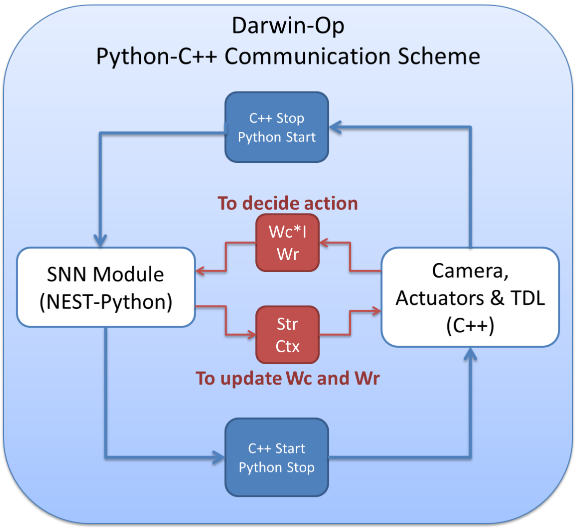

Figure 4.3: Model is coded in two parts on the humanoid robot which are the module responsible for action selection and the module responsible for parameter adaptation to accomplish learning. The action selection model is coded in Python environment using NEST simulator for the spiking neural network part [28]. Getting sensory input and actuation part is coded in C++ including learning. This diagram shows the communication scheme of the communication between two environments.

association to a new action. This task is achieved by updating the connections between sensory stimuli, cortex and striatum as explained in Section 3.1.

To show success of the model in real time learning task, two experiments are realized on humanoid robot. In the first experiment, the sensory inputs are associated to the desired actions in sequence and then the previously associated pair is rearranged. In the second experiment, the sensory inputs are presented to the robot in random order and association time and the strength of the connections are investigated.

Results of the first experiment can be seen in Figure 4.4. In addition, raster plot of this experiment is given in Figure 4.5. Raster plot shows the spike activity of channels with respect to time. In Figure 4.5, the spike activities of the first channel and third channel are given at the time intervals of $550-750^{th}$ ms and $3550-3750^{th}$ ms. The upper raster plots of Figure 4.5, show the activity between 550 and 750^{th} ms and the lower two raster plots show the activity between 3550 and 3750^{th} ms. The y axis of the raster plots show the IDs of neurons that fire. The x axis shows the time. The points indicate the spike at the related time. The bars of raster plots show the average firing rate of that channel at the related time. It can be followed from the upper raster plots of Figure 4.5 that the activity of two channels are almost same at the beginning of the experiment.

a)
b)
c)

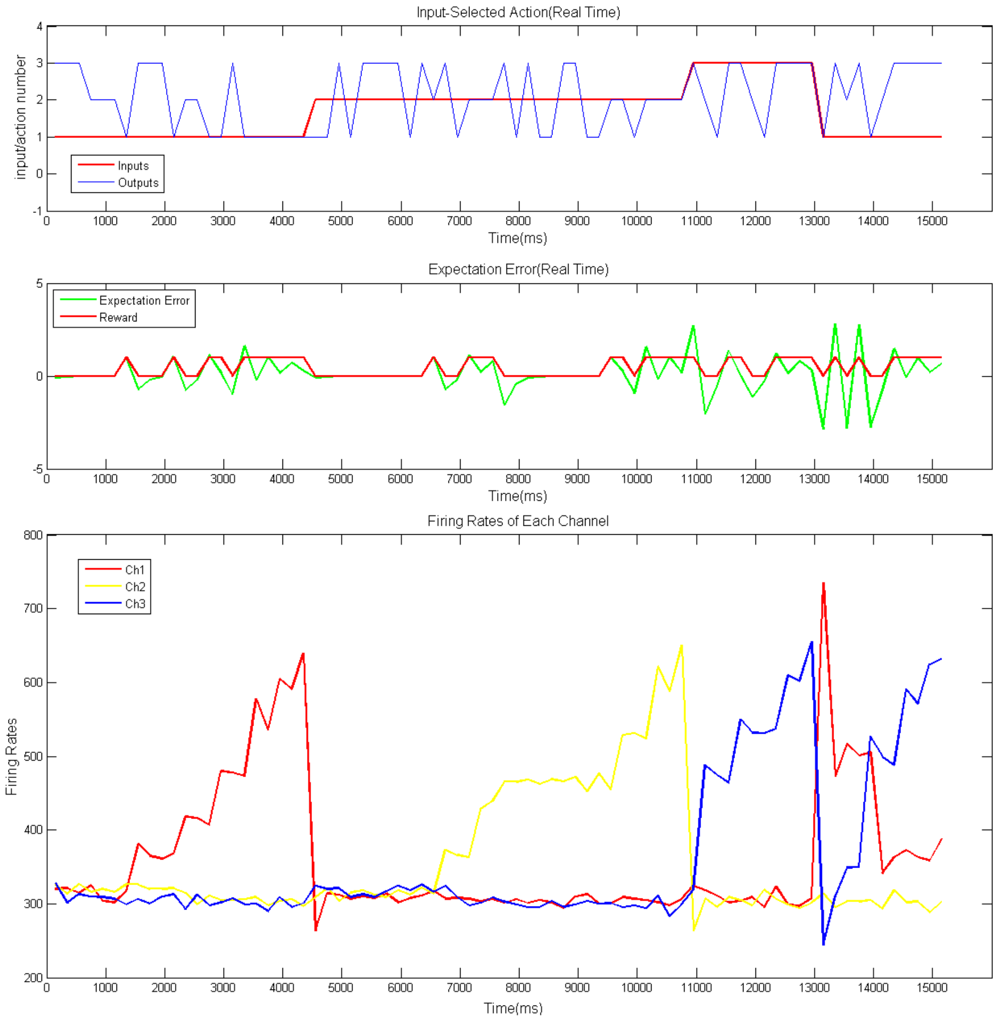

Figure 4.4: a (upper figure): The selected actions (blue line) and the sensory inputs (red line). The first stimulus is red color, the second is yellow color and the third is blue color. b (middle figure): Reward (red line) and expectation error (green line). c (lower figure): Average firing rates of cortex channels. The simulated time of the spiking neural network last 15150*ms* for this experiment, but it takes 45 minutes in real time, real time factor (the proportion of simulation time to the real time) of which is approximately %0.6.

However, the lower raster plots indicate that the first channel fires more than the third channel and the third channel fires even less than beginning. The reason of this is the connections between the sensory input and the first channel are strengthened during the 3000 *ms* with reward and the connections between the stimulus and the third channel are weakened. The Figure 4.4-a shows the presented input, which are red, yellow and blue colors respectively, and the channel of the selected action. The sensory input and the selected actions are indicated with red and blue lines, respectively. The Figure 4.4-b shows the expectation error, green line, and reward, red line. The Figure 4.4-c shows the average firing rates of the each channel in the cortex and red, yellow and blue lines indicate the channels respectively. Through the experiment, the spiking neural network is simulated 200*ms* for each sensory input. The average firing rates of cortex channels are calculated over the spikes in this 200*ms* time interval. The simulated time of the spiking neural network last 15150*ms* for this experiment, but it takes 45 minutes in real time, real time factor is approximately %0.6. However, showing the color stimulus and reward to the robot is included in the elapsed time of real time experiment while it is not included for the simulation. That's why the real time factor is lower than the expected value. In addition, this is due to the processor inside the robot that is not suitable for a spiking neural network simulation in real time. During the experiment,

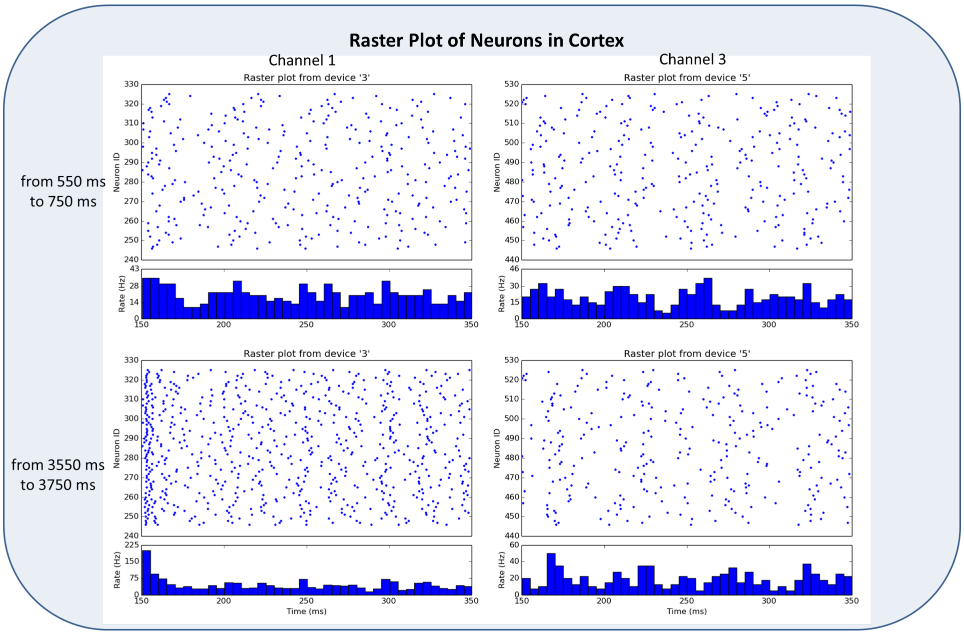

Figure 4.5: Raster plot of the first and third channels during the experiment. The upper raster plots show the activity between 550 and 750^{th} ms and the lower two raster plots show the activity between 3550 and 3750^{th} ms. The y axis of the raster plots show the IDs of neurons that fire. The x axis shows the time. The points indicate the spike at the related time. The bars of raster plots show the average firing rate of that channel at the related time. The connections between the first channel and the stimulus are potentiated and the connections between the third channel and the stimulus are depressed. So, the firing activity of the first channel is more than the firing activity of the third channel.

the sensory stimuli are first associated with the desired actions which are the first input (red color) to the first action (channel 1), the second input (yellow color) to the second action (channel 2) and third input (blue color) to the third action (channel 3) as given in Figure 4.4-a. At the beginning, the humanoid robot selects actions randomly, since there is no winner between cortex channels until one of the average firing rates reaches to a certain value. After the colors are presented, the expectation error remains zero as far as the first reward to that input is given. This situation is given in Figure 4.4-b,c between the time intervals, $0-1000ms$ and $5000-6500ms$.

When the red color (first stimulus) is presented, the humanoid robot selects a random action until the average firing rate of the first channel reaches to a certain value. Until the first reward, the expectation error remains zero; this is why the connections and firing rates of channels remains same. This situation can be followed from Figure 4.4-c between the time intervals, $0-1000$ ms and $5000-6500$ ms. In Figure 4.6, the evolution of connections between sensory inputs (I) and cortex channels (Ch) through the experiment can be followed. At the beginning, the connections have a random value close to zero and evolve to values which build the associations between sensory inputs and desired actions in the way that the expectation error decrease to zero. After all sensory stimuli are associated with the actions, the first sensory stimulus

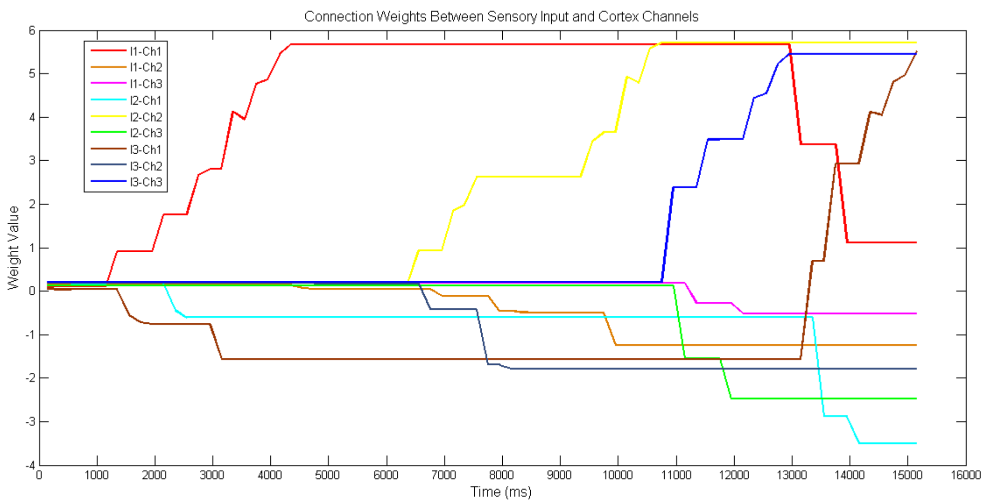

Figure 4.6: The evolution of connections between sensory inputs (I) and cortex channels (Ch) through the first experiment.

is reassociated to the third action at the end of the experiment to show the realization of rearrangement of associations. After 13000^{th} ms the first sensory stimulus is associated to the third action by rewarding selection of the third action instead of the first. Therefore, the connections between the first stimulus and the first action decrease while the connections between the first stimulus and the third action increase (Figure 4.6). In Figure 4.4-c, it is seen that the average firing rate of the third channel increases due to the change in the connections. However, the connections between the first input and the first channel is still higher than the value at the beginning. So, they can be reassociated more easily considering the association at the beginning, which is also compatible to the reinforcement learning aspect.

As a second experiment, the sensory inputs are presented in a random order seen in Figure 4.7. In the second experiment, the first input associated to the first action and so on. After all associations are accomplished, the associated action of the first input is changed to be third action. All processes are the same as the first experiment, but the orders of presented sensory stimulus are random. The average firing rates during the second experiment can be seen in Figure 4.8. On the 11000^{th} ms all inputs are associated to the desired actions and the rearrangement of association of the first sensory stimulus begins after then. The second experiment is terminated after 15150 ms, since the rearrangement is accomplished. In total, the task is completed in approximately same time interval in both of the experiments even the learning in the second experiment is realized in random order. The evolution of connections is presented in Figure 4.9 for the second experiment. Since the actions are selected randomly, when there is no winner, some of the connections are depressed in proportion to the expectation error because of not having expected reward. This situation happens for the connections between I3-Ch1 (green line), I2-Ch3 (cyan line) at the beginning of the experiment. The I1-Ch1 connection (red line) also decreases after 10500^{th} ms, but to a value close to one which makes a further association easier.

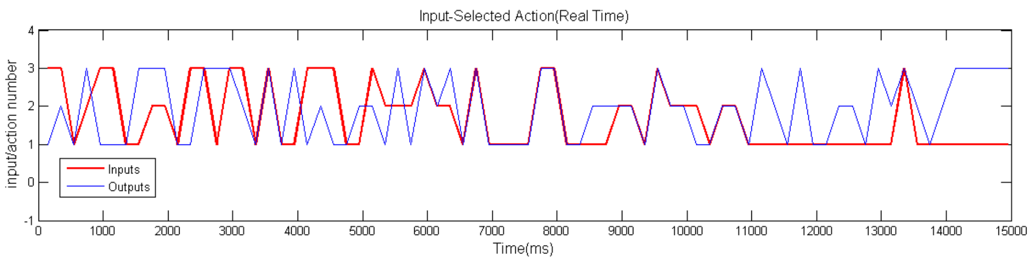

Figure 4.7: The selected actions (blue line) and the sensory inputs (red line) of the second experiment. The first input is red color, the second is yellow color and the third is blue color. The sensory stimuli are presented in random order.

4.4 Conclusion

In this part of the study, learning to build associations between the sensory inputs and actions are realized using point neuron approach in relation with mass model. Through

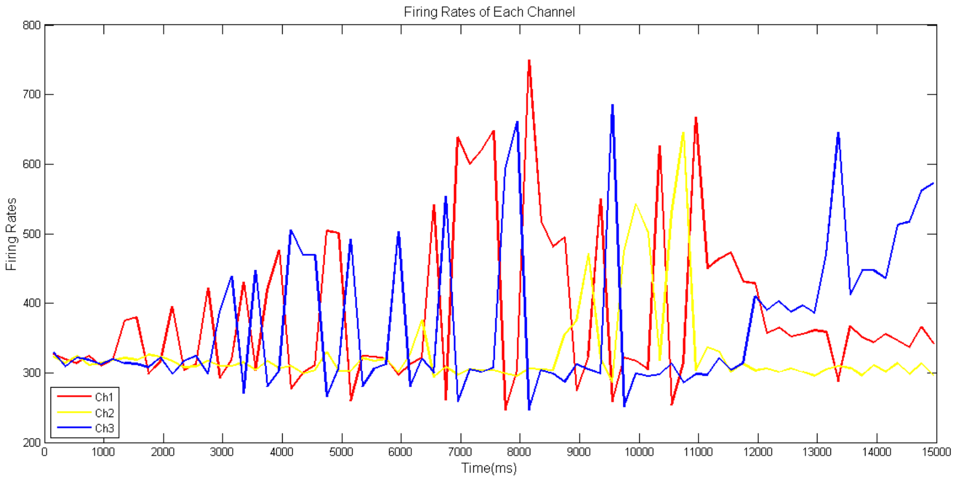

Figure 4.8: Average firing rates of cortex channels through the second experiment.

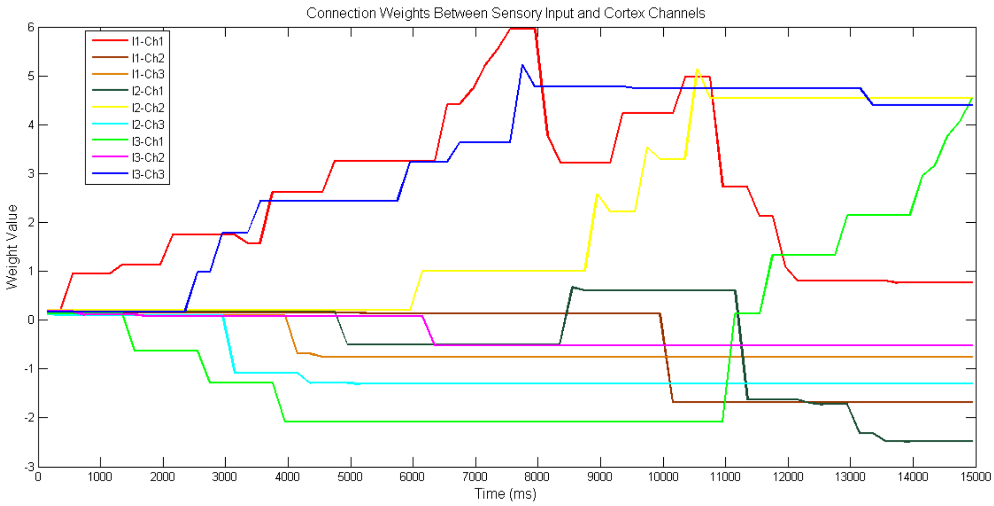

Figure 4.9: The evolution of connections between sensory inputs (I) and cortex channels (Ch) through the second experiment.

the task, an association of the visual sensory inputs to predefined actions are built up. The computational model in [13] is extended for this task with point neurons inside cortex and reward modulated connections. Since embodiment of the computational models of neuronal circuits is an emerging way of investigating brain organisation, an environment for the realization of action selection circuit and learning is built to simulate the computational model in real time. Neural structures of the basal ganglia and the cortex are modeled in a simple way to decrease the computation need through the task, since the aim is to investigate the applicability of such model on the humanoid robot platform in real time. Despite these kind of humanoid robot platforms have high mobility abilities, they have low computation abilities for embodied simulation of neural circuits. As a result of this, the simulation of 300 point neurons and the dynamical system model has a $0.6-1\%$ real time factor. One of the important aspect for the embodiment is that since the point neurons in cortex are in relation with the basal ganglia and thalamus structures which are modeled as mass model, the neuron parameters doesn't need to be optimized for a specific task. Therefore, despite the lack of model reality and detail in the computational models of BTC loop for action selection, this simple approach is sufficient from modeling aspect to show the action selection behaviour on cortex in real time applications.

5. ASSOCIATION TASK USING CTX SNN AND STDP

In Chapter 4, a step is taken toward obtaining a biologically realistic model for reward based learning by modeling the cortex with point neurons. Even though, the neural structure is realized by point neurons, still TDL method is used for learning. Here, as learning rule, a synaptic plasticity rule will be implemented. Since, in Chapter 4, the realization of point neuron model on humanoid robot is done, and the time limit for online learning has been discussed, here the results will be obtained only as simulations.

In the sequel, first definition of spike time dependent plasticity will be given, then its implementation for the cortex model in Chapter 4 will be realized.

5.1 Spike Timing Dependent Plasticity (STDP)

Learning and memory in the brain are usually associated with synaptic behaviour and synaptic adaptation [30]. One of the essential idea on learning is explained by Donald Hebb which is called Hebbian learning in 1949. Hebbian learning postulates that insistent spike behaviour of postsynaptic neuron just after the spike behaviour of presynaptic neuron increases the synaptic strength due to biological processes forming the synapse between the two neurons. Despite the potentiation of synapse is explained by Hebb, a rule for depression of the synapse is not defined explicitly so giving rise to permanently increasing synaptic strength. Spike timing dependent plasticity (STDP) is a version of Hebbian learning that considers the temporal differences of spike activity in the presynaptic and postsynaptic neurons [31]. Unlike the Hebbian learning, STDP considers the depression besides the potentiation. Not first but leading experiments in [32] and [33] showed that the repeated activation of presynaptic neuron before the activation of postsynaptic neuron in a certain time interval potentiates the synapse between two neurons and the reverse situation in a certain time interval causes depression in synapse. The former is called as long term potentiation (LTP) and the latter is called as long term depression (LTD) [31]. The importance of this model comes from the biological plausibility and usability to explain the learning and memory process of brain that is supported by experiments [31].

The modulation of synapse strength is related to the activation of the presynaptic and postsynaptic neurons. However, Izhikevich (in [29]) links STDP with a reward signal to explain conditioning with a point neuron and reinforcement learning point of view. This approach combines the two explanation about learning process in the brain. Izhikevich models the reward modulated STDP with Equations 5.1, 5.2 and 5.3.

$$\dot{c} = -c/\tau_c + STDP(\tau)\,\delta\left(t - t_{pre/post}\right) \tag{5.1}$$

$$\dot{s} = cd \tag{5.2}$$

$$\dot{d} = -d/\tau_d + DA(t) \tag{5.3}$$

In these equations, d stands for extracellular dopamine level, δ is the Dirac function with respect to the time difference of neuron activities, $STDP(\tau)$ is the STDP function that is shown in Figure 5.1. This STDP function determines the scale of potentiation or depression with respect to the time interval between the firing activity of two neurons. c is defined as "eligibility trace" which indicates that the synapses are eligible to be modulated. All τs are time constants of related variable. s is the synaptic strength between two neurons. $DA(t)$ indicates the baseline level of dopamine concentration. The time interval for both potentiation or depression between two neurons is considered as 50 ms. Closer timing gap between two neurons means more potentiation or depression.

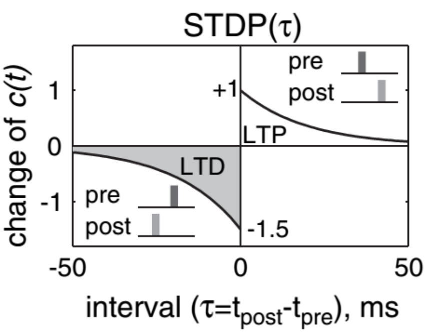

Figure 5.1: STDP function that is retrieved from [29].

The modulation of the synaptic strength is explained as shown in the Figure 5.2. Here reward is modeled as the dopamine concentration. After successive firing of pre and

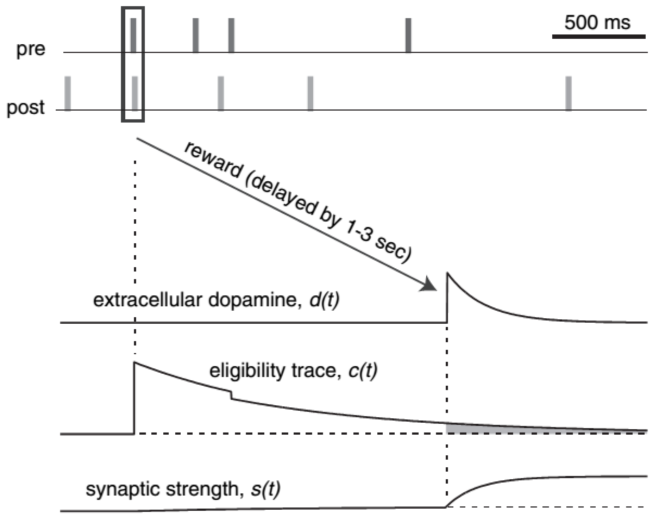

Figure 5.2: The modulation of synaptic strength that is retrieved from [29].

post neurons given in the box in Figure 5.2, an exponentially decreasing eligibility trace occurs. After specified delay time that changes from 1 to 3 seconds, reward is given to the system, and the concentration of dopamine increases and the modulation on the synaptic strength is realized as seen from the lower line of the Figure 5.2. The synaptic strength increases in this situation since activation of the pre neuron occured just before the activation of the post neuron. In this case the result of STDP function is positive since τ is greater than zero as seen in Figure 5.1. So, LTP and LTD are realized by applying the same approach.

5.2 Implementation of STDP into SNN Based Cortex Model

In this chapter we utilized the color association task in the same way as explained in Chapter 2. However, the action selection model in the task is designed different from the models that are utilized in the previous chapters. The model considered here is presented in Figure 5.3. As it can be followed from the figure, a model of cortex with STDP learning rule is given. With this new cortex model with intrinsic synaptic plasticity, the association between, sensory inputs and their representations in the cortex is built up without defining matrix W_c and using learning rules adapted from a machine learning method. So, a biologically-plausible model of association building has been set up. In this model R, Y and B circles represent the sensory inputs which are modeled as "poisson generators" that spikes with poisson distribution. They project the sensory information to the input neurons which are modeled as "izhikevich regular spiking neurons". There are 20 neurons in each channel of input neurons. The rate of spiking activity for sensory inputs is 12 Hz when there is a high sensory information and 3 Hz when there is a low sensory information. Cortex is modeled same as the model that is explained in section 4. The information coming from sensory inputs to input neurons are projected into cortex neurons through STDP modulated synapses.

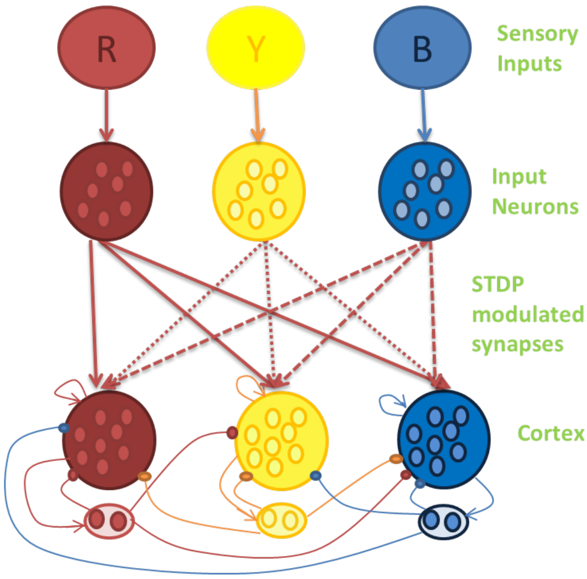

Figure 5.3: The computational model considered to utilize STDP modulated synapses.

Instead of implementing the Equations 5.1, 5.3 and 5.2, the NEST library is utilized for the neuron and synapse simulations as utilized for the cortex model that is explained in section 4. In this part of study, "stdp_dopamine_synapse" model of NEST ([34]) is used as the STDP modulated synapse model instead of the W_c matrix of previous models since STDP is a more biologically plausible implementation of reinforcement learning. The parameters of the model are given in Table 5.1. The w_{max} and w_{min} parameters indicate the maximum and minimum connection weights that a synapse can have. The τ_c and τ_d represent the time constants that are explained in section 5.1. τ_{pre} and τ_{post} stand for the time constants of STDP function that is given with Figure 5.1. The *baseline* value is the minimum level of dopamine concentration that takes part in Equation 5.3 as $DA(t)$. *initialweights* is the initial values of synaptic connection of the STDP modulated synapses.

Table 5.1: STDP connection parameters.

w_{max}	w_{min}	τ_c	τ_d	τ_{pre}	τ_{post}	*baseline*	*initialweights*
20.0	3.0	500 * ms	800 * ms	50 * ms	50 * ms	0.01	9.0

The action selection is realized with this model as it is explained in the "Stimulus-Response Instrumental Conditioning" part in [29]. In a certain time interval, the number of spikes that are counted are due to the stimulus applied to. If the channel, which has the greatest number of spikes, is the desired one, then reward is given to the system. The reward is given as a step current to a group of neurons which excites neurotransmitter amount of STDP modulated synapses. This neurotransmitter amount excites the level of extracellular dopamine that is explained in [29] and [34].

5.3 Results

The considered task is similar to the task that are handled in Chapters 3 and 4. At first, the red color will be associated to the first channel, the yellow color will be associated to the second channel and the blue color will be associated to the third channel. After all associations, the red color will be reassociated to the second channel, the yellow color to the third channel and the blue color to the first channel. In this way, the association and rearranging the associations will be handled.

The coincidence of spiking activity of presynaptic and postsynaptic neurons effects the modulation of synaptic strength as explained in section 5.2. This process can be followed from Figures 5.4 and 5.5. These two figures show the beginning of an experiment. In Figure 5.4, the spike activity of input and cortex neurons are shown. Only the red color stimulate the network in this time interval. The spike counts of input neurons of first channel vary since the input neurons are stimulated by poisson generators. After stimulation of input neurons, cortex neurons are stimulated by the input neurons. The synaptic weights between the channels of input neurons and cortex neurons are determined randomly in neighborhood of a mean value. Therefore, the first channel of input neurons can stimulate all channels of cortex at the beginning of the experiment as seen in the lower graph of Figure 5.4. The coincident firings of neurons in a certain time interval generate eligibility traces and in that time interval the synapse weights can be updated. In Figure 5.5, the middle graph shows the mean eligibility values of the synapses. The red, yellow and blue lines show the eligibility values of the synapses between the first channel of input neurons and the first, second and third channels of cortical neurons, respectively. In Figure 5.4, after the 10000^{th} ms Ch1 and Ch3 of cortex generate almost the same amount of spikes, however eligibility of synapses to Ch1 is greater than Ch3 in Figure 5.5. The reason of this may be the distance between the spike times of input Ch1 and cortex Ch3. So, timing is also important even the spike amount is high. The first channel of cortex wins after the 7000^{th} ms and since this is the desired situation, reward is given to the system which can be seen from upper graph of Figure 5.5 that shows the change of dopamine level. Since dopamine level is under the baseline level which is 0.01, the weights decrease. The lower graph of Figure 5.5 shows the depression of mean weights that mean the mean value of strength of all synaptic connections. The mean weights decrease until the given reward which keeps the dopamine level at the baseline.

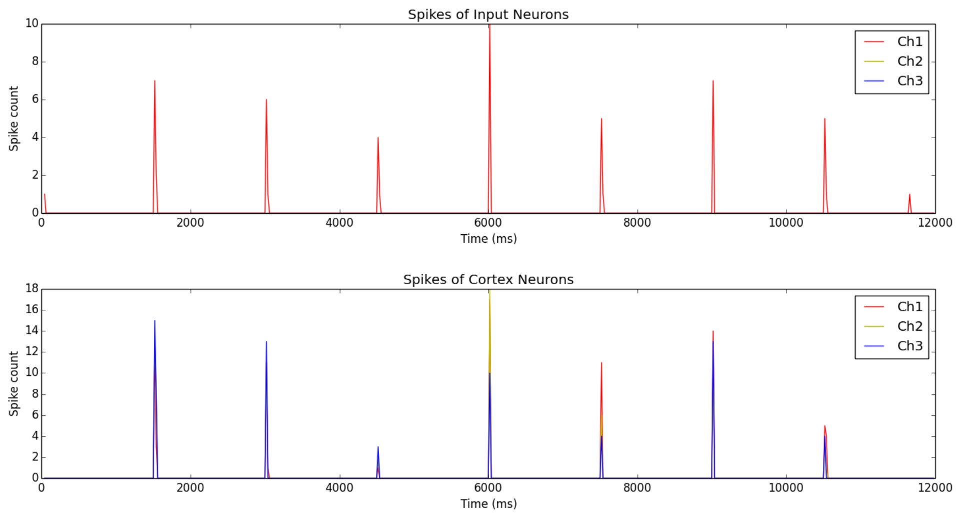

Figure 5.4: Spike activities of input and cortex neurons at the beginning of an experiment.

A complete simulation can be followed from Figures 5.6 and 5.7. In Figure 5.6, upper graph shows the spike activity of input neurons, the middle graph shows the spike activity of cortical neurons and the lower one shows dopamine level. In the upper figure of Figure 5.6, the red, yellow and blue colors indicates the first, second and third channel of inputs, respectively. The colors are same for the middle figure of Figure 5.6, in which red, yellow and blue colors represent the first, second and third channels of cortex. The given stimuli are changed after 1 000 000 ms. It can be seen from the Figure 5.6 that at the beginning of each new stimulus, the spike activity of desired channel in the cortex is low. After a while the connections strengthen and the spike activity of related cortex channel increases as a result. At the first 3 000 000 ms, the first associations of stimuli are realized. The rearrangement of associations are made

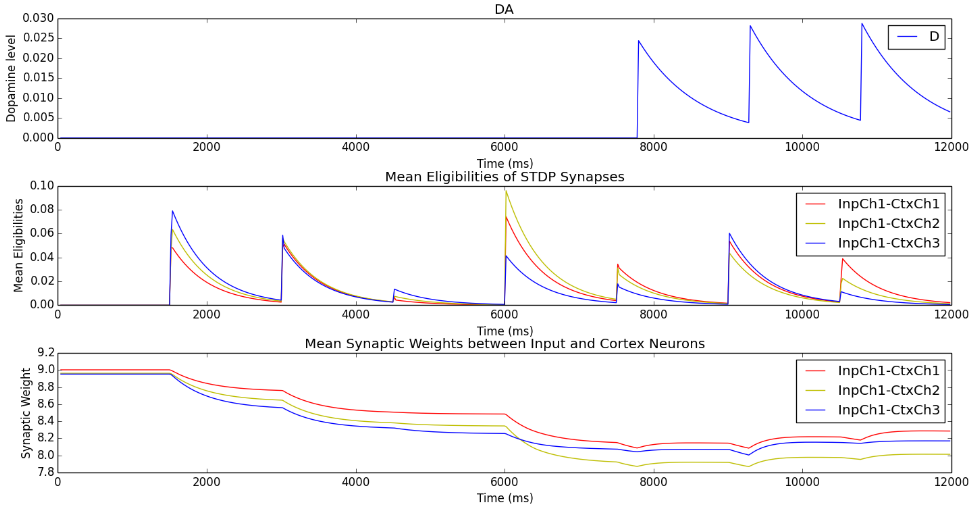

Figure 5.5: Change of dopamine (DA) level, eligibility traces and mean synaptic weights in time at the beginning of an experiment.

at the last 3 000 000 *ms*. The change of the mean synaptic weights are given in Figure 5.7. Eligibilities are not included to make explanation simpler. The upper, middle and lower graphs of Figure 5.7 show the synaptic weights of connections between the first, second and third channels of inputs and the channels of cortex, respectively.

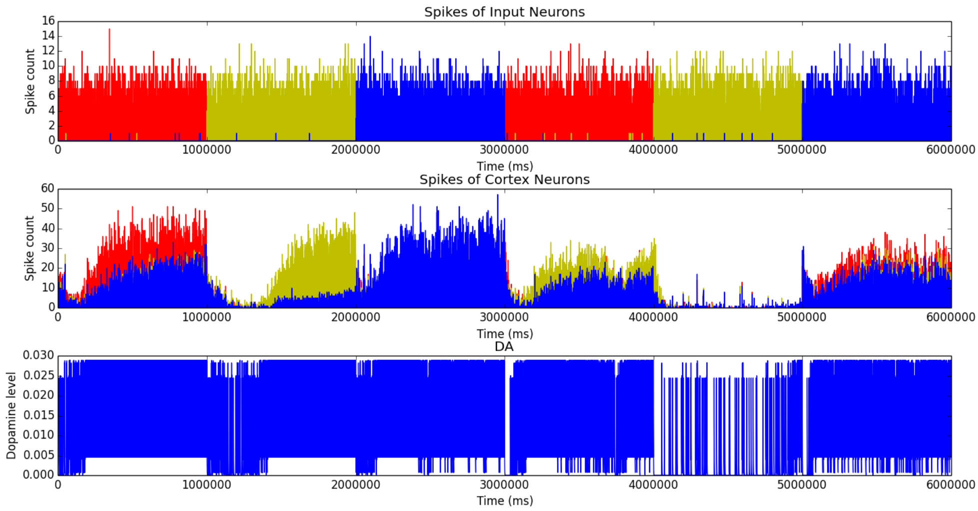

Figure 5.6: Spike activities of input and cortex neurons with dopamine level at the first experiment. Red, yellow and blue colors indicate activities of the first, second and third channels of related graph, respectively.

All weights decrease to a level at the beginning of a given stimulus before LTP. The reason of this is a channel has to suppress the other channels in order to win. So, the weights decrease to a level until the desired channel at the cortex wins and gets reward successively. The associations built between the first input channel and first cortex

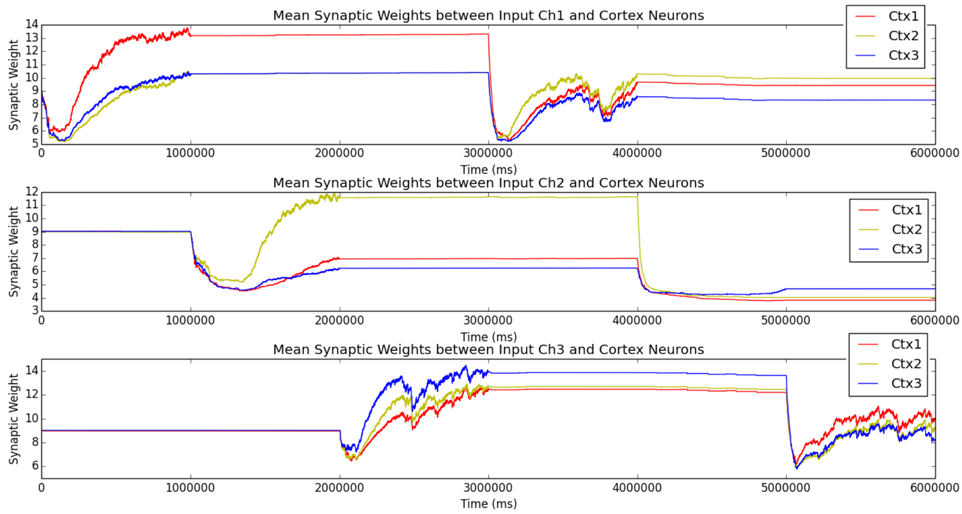

Figure 5.7: Changes of synaptic weights between input channels and cortex channels at the first experiment.

channel and between the second input channel and second cortex channel display a big difference, since strength of the other synapses decrease more at the beginning of the given stimuli. However, as it can be followed from the lower graph of Figure 5.7, the connections between the third input channel and cortex don't decrease lower than 6. So, even the desired synapses are stronger than the others, there is not a big difference for the synaptic weights of the third channel. After the 3 000 000th *ms*, the reassociations begin with 1 000 000 *ms* intervals. The synaptic weights outgoing from the first and third manages to be reassociated as followed from the change of cortex spike activity (seen from the Figure 5.6, middle graph). Looking to the middle graph of Figure 5.7, the depression is managed on the reassociation process because of the lack of reward, but potentiation is not big for the synapses between the second input channel and third cortex channel. At the 5 000 000th *ms*, since the slope of the blue line is strongly positive, it can be explained that the synaptic connections are potentiated (Figure 5.7, middle graph), and the strength of other synapses remain lower.

At the second experiment, the same task is repeated as it can be followed from the Figures 5.8 and 5.9. The first associations are successful (until 3 000 000th *ms*) and this can be followed from the given reward and spike activity of cortex neurons. The reassociations are also successful, but the reassociation between the first input channel and the second cortex channel is potentiated a little at the end of the process of red color reassociation. The continuation of stimulating with red color would provide

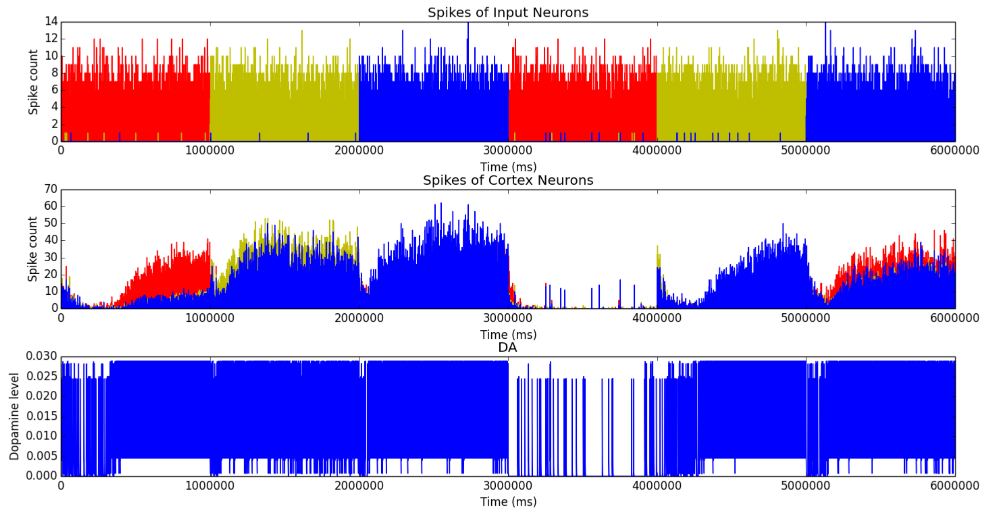

Figure 5.8: Spike activities of input and cortex neurons with dopamine level at the first experiment. Red, yellow and blue colors indicate activities of the first, second and third channels of related graph, respectively.

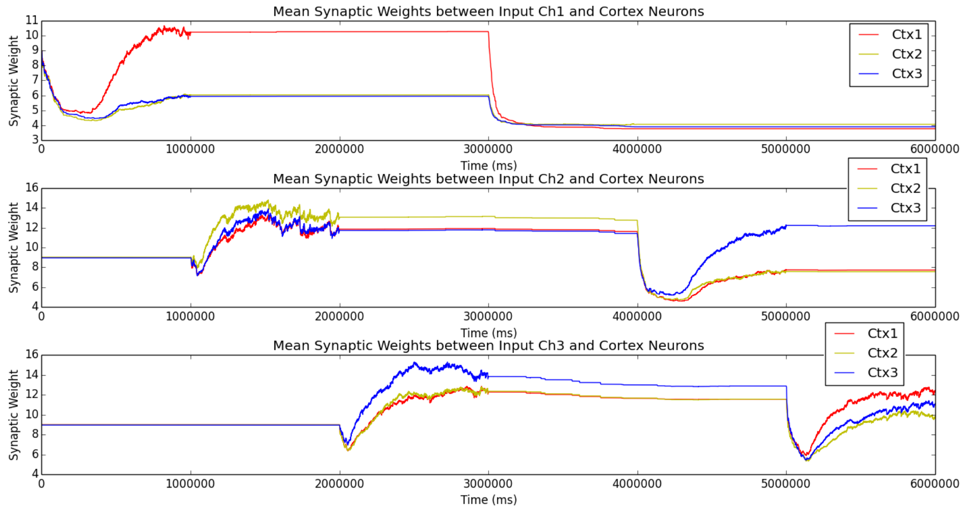

Figure 5.9: Changes of synaptic weights between input channels and cortex channels at the first experiment.

more potentiation after a while. It is clearly seen that, to potentiate specific synaptic weights, the mean synaptic weights should decrease below 6. Otherwise, the synaptic weights will trace the same pattern as it happened in the middle graph of Figure 5.9. Decreasing below 6 helps the inhibitory neurons of cortex channels to suppress the other channels. So, this suggests that the inhibitory network doesn't work well enough for all situations and should be investigated more. However, this is a different study topic on itself like the investigation of the BTC model in section 3.2.

The aim of this section was to investigate the STDP process and to present an STDP network for learning instead of the TDL rules which is also realistic in theory but not in computational way. That's why the network is reduced to only input and cortex parts comparing to the BTC models in Chapters 4 and 3. The structure that is minimized is compensated with the dopamine process.

6. CONCLUSIONS AND RECOMMENDATIONS

In the thesis, computational models of action selection are implemented on humanoid robot platform, Darwin-OP. The computational models are handled from the dynamical system approach to the point neuron approach to reach the realistic plausibility. In Chapter 3, the dynamical system model of BTC is analyzed. The model is investigated to check whether the results obtained at the end of learning process are generic, since the parameters of the dynamical system are changed and the overall behavior of the system has been completely differentiated within the learning process. So, the parameter values and their meanings in the sense of behavior are investigated by tampering the connections. In addition, the model is implemented to Bioloid robot platform to show its usability. In Chapter 4, the same model is handled and its cortex part is changed to SNN model instead of mass model. In that way, biologically realistic neuron model is utilized at least for a part of the computational model and the model used had different scales together. In this chapter, the convenience of the robot environment, Darwin-OP, for the neuron model approach is shown. The aim of this study is to show the possibility of modeling the entire network with point neuron models. However, adapting a more realistic learning method is necessary to make use of SNN model for BTC circuit. Due to this, the STDP approach is considered and implemented in cortex where SNN is used to realize the action selection in Chapter 5. Though, the scale of SNN is small, still satisfactory results are obtained.

In Section 3.2.1, the mass model is investigated from the initial conditions aspect. The effect of initial conditions is hard to investigate on brain because of its distributed and complex structure. However, this investigation is meaningful from the perspective of the first encountering of an agent with a new environment. As a result, computational model can be useful for anticipating the behavior of an agent in different environmental conditions and also it is successful to regulate the computational stability on learning. Another result is when different input values are presented to the model, the success rate on learning and the elapsed time changes. This shows that the strength of sensory inputs also have effect on learning. The sequence of sensory inputs also affects the learning from the perspective of both the time and the success. When the inputs are presented in random order, the learning time increases, but the learning is stronger. Though, this is another research topic, this may be related to the phenomenon where learning language for babies is harder when they live in a multi-lingual environment, but they can speak more fluently all the languages in the environment after the learning phase is accomplished. In addition, the model manages to select the right action after learning is accomplished as explained in Section 3.2.2.

As it can be followed from results of the Section 3.2.3, the strength of transferred information has an influence on learning. When S_{max} has the maximum value, which means all the sensory information is transferred to the substructures, this causes zombie situation and the model cannot change its behaviour for the new sensory inputs. In addition, the base level of dopamine in the model has a huge impact on learning process. The base level of dopamine affects ability of selecting only one action at a time. Another result is that high thalamus activity on cortex decreases possibility of being successful on selecting right action. This also indicates a kind of hiperactivity situation, which impairs the action selection process. Tampering the other connections in the model decreases the learning success for all parameter values since the information transfer to the substructures is damaged.

From the results of Chapter 4, it can be deduced that when an association is built between a sensory input and an action, it is easier to make a reassociation between them even if the first association is destroyed for another association. Another result coming from the Chapter 5 is that the activity of an undesired cortex channel has to be lower than a certain activity to be able to learn the task in a more stable way. All these behavioural results of computational models show that it is possible to make inference from these models to related real processes. In addition, it is possible to realize massive numbers of experiments. Although, these models are far from showing the exact process of brain, the results indicate that these models can help to steer the examinations for real experiments. Also, since the robots provide mobility, this models can handle more complex tasks in an easier way instead of using animals for all of the experiments.

From the embodiment aspect, considering only the BTC circuit is not sufficient to model the mobility tasks. Especially cerebellum has to be included for such a model with hippocampus for the perception of environment, spatial navigation, memory, attention and motor control. Since the environment in real world will not be an ideal one, the role of other brain structures becomes significant to deal with different ground properties such as grainy, gravelled, slippery, etc. which are important for practical reasons. In addition, the sensory inputs have to be more in number to provide additional information to the other structures in the brain. Though, entire network has to be set up to provide a realistic experiment, this thesis is one of the steps to complete the pieces of the entire puzzle.

The implementation of association task in Chapter 3 is partly presented as poster presentations ([35], [36]) at 12^{th} National Neuroscience Congress in Turkey and at International Workshop on Autonomous Cognitive Robotics in Scotland. Early results on changing the cortex part of BTC model with SNN structure and implementation of this model on Darwin-OP that takes part in Chapter 4 are presented as a poster presentation [37] at Bernstein Conference 2014 in Germany. The expanded study of Chapter 4 is accepted to be presented [38] at The International Joint Conference on Neural Networks (IJCNN) 2015 in Ireland. Lastly, the cortex model and the STDP learning approach in Chapter 5 is used to model the sensory and motor cortex and generate the learning part in the study [39] that is presented at the 23^{th} Signal Processing and Communications Applications Conference in Turkey.

REFERENCES

[1] **Kudoh, S.N.**, **Ito, H. and Hayashi, I.** (2012). Neurorobot VitroidA living test model for embodiment brain research, *Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference on*, IEEE, pp.1472–1475.

[2] **Seepanomwan, K.**, **Caligiore, D.**, **Cangelosi, A. and Baldassarre, G.** (2015). Generalisation, decision making, and embodiment effects in mental rotation: A neurorobotic architecture tested with a humanoid robot, *Neural Networks*.

[3] **DeLong, M.R. and Wichmann, T.** (2007). Circuits and circuit disorders of the basal ganglia., *Archives of neurology*, ***64***(1), 20–24.

[4] **Alexander, G.E. and Crutcher, M.D.** (1990). Functional architecture of basal ganglia circuits: neural substrates of parallel processing, *Trends in neurosciences*, ***13***(7), 266–271.

[5] **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.

[6] **Prescott, T.J.**, **Montes González, 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.

[7] **O'Reilly, R.C. and Frank, M.J.** (2006). Making working memory work: a computational model of learning in the prefrontal cortex and basal ganglia, *Neural computation*, ***18***(2), 283–328.

[8] **Takakusaki, K.**, **Tomita, N. and Yano, M.** (2008). Substrates for normal gait and pathophysiology of gait disturbances with respect to the basal ganglia dysfunction, *Journal of neurology*, ***255***(4), 19–29.

[9] **Schultz, W.**, **Dayan, P. and Montague, P.R.** (1997). A neural substrate of prediction and reward., *Science (New York, N.Y.)*, ***275***(5306), 1593–1599.

[10] **Baladron, J. and Hamker, F.H.** (2015). A spiking neural network based on the basal ganglia functional anatomy, *Neural Networks*, ***67***, 1–13.

[11] **Chersi, F.**, **Mirolli, M.**, **Pezzulo, G. and Baldassarre, G.** (2013). A spiking neuron model of the cortico-basal ganglia circuits for goal-directed and habitual action learning, *Neural Networks*, ***41***, 212–224.

[12] **Fiore, V.G.**, **Sperati, V.**, **Mannella, F.**, **Mirolli, M.**, **Gurney, K.**, **Friston, K.**, **Dolan, R.J. and Baldassarre, G.** (2014). Keep focussing: striatal dopamine multiple functions resolved in a single mechanism tested in a simulated humanoid robot, *Frontiers in psychology*, ***5***.

[13] **Şengör, N.S.**, **Karabacak, O. and Steinmetz, U.** (2008). A computational model of cortico-striato-thalamic circuits in goal-directed behaviour, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, ***5164 LNCS***(PART 2), 328–337.

[14] **Denizdurduran, B. and Sengor, N.S.**, (2012). Learning how to select an action: A computational model, Artificial Neural Networks and Machine Learning–ICANN 2012, Springer, pp.474–481.

[15] **Denizdurduran, B. and Sengor, N.S.** (2012). A Realization of Goal-directed Behavior-Implementing a Robot Model based on Cortico-Striato-Thalamic Circuits., *ICAART (1)*, pp.289–294.

[16] **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 of London B: Biological Sciences*, ***362***(1485), 1573–1583.

[17] **Tanji, J. and Shima, K.**, (1994), Role for supplementary motor area cells in planning several movements ahead.

[18] **Erçelik, E.** (2013). Learning To Select an Appropriate Action: Implementation on Bioloid Robot, *B.Sc. Graduation Project*, Istanbul Technical University Electric and Electronics Faculty, Electronics Engineering Programme, graduation Project.

[19] **Erçelik, E.**, **Denizdurduran, B. and Şengör, N.S.** (2013). A Bioloid Robot Implementation of Computational Basal Ganglia Circuit, 11. National Neuroscience Congress, Türkiye.

[20] **Sutton, R. and Barto, A.** (1998). Reinforcement learning: An introduction.

[21] **Denizdurduran, B.** (2012). Learning How To Select An Action: From Bifurcation Theory To The Brain Inspired Computational Model, *M.Sc. Thesis*, Istanbul Technical University Institute of Science and Technology.

[22] **Grossberg, S.**, (1982). How does a brain build a cognitive code?, Studies of Mind and Brain, Springer, pp.1–52.

[23] **Crick, F. and Koch, C.** (2003). A framework for consciousness, *Nature neuroscience*, ***6***(2), 119–126.

[24] **Alexander, G.E.**, **Crutcher, M.D. and DeLong, M.R.** (1989). Basal ganglia-thalamocortical circuits: parallel substrates for motor, oculomotor," prefrontal" and" limbic" functions., *Progress in brain research*, ***85***, 119–146.

[25] **Frank, M.J.** (2006). Hold your horses: a dynamic computational role for the subthalamic nucleus in decision making, *Neural Networks*, ***19***(8), 1120–1136.

[26] **Dominey, P.**, **Arbib, M. and Joseph, J.P.** (1995). A model of corticostriatal plasticity for learning oculomotor associations and sequences, *Journal of cognitive neuroscience*, ***7***(3), 311–336.

[27] **Izhikevich, E.M.** (2003). Simple Model of Spiking Neurons, *Neural Networks*, ***14***(6), 1569–1572.

[28] **Gewaltig, M.O. and Diesmann, M.** (2007). NEST (neural simulation tool), *Scholarpedia*, ***2***(4), 1430.

[29] **Izhikevich, E.M.** (2007). Solving the distal reward problem through linkage of STDP and dopamine signaling, *Cerebral Cortex*, ***17***(10), 2443–2452.

[30] **Hebb, D.O.** (1949). *The organization of behavior: A neuropsychological theory*, New York: Wiley.

[31] **Sjöström, J. and Gerstner, W.** (2010). Spike-timing dependent plasticity, *Scholarpedia*, ***5***(2), 1362, revision #151671.

[32] **Markram, H.**, **Lübke, J.**, **Frotscher, M. and Sakmann, B.** (1997). Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs, *Science*, ***275***(5297), 213–215.

[33] **Bi, G.q. and Poo, M.m.** (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type, *The Journal of neuroscience*, ***18***(24), 10464–10472.

[34] **Url-1**, *<http://www.nest-simulator.org/cc/stdp_dopamine_synapse/>*, date retrieved 2015-11-07.

[35] **Erçelik, E. and Şengör, N.S.** (2014). Uyaran-Eylem İlişkilendirme Ödevinin İnsansı Robot Üzerinde Gerçeklenmesi, 12. National Neuroscience Congress, Türkiye.

[36] **Erçelik, E. and Şengör, N.S.** (2014). Implementation of Matching Stimulus-Movement Experiment on a Humanoid Robot, *International Workshop on Autonomous Cognitive Robotics.*, Stirling, Scotland.

[37] **Erçelik, E. and Şengör, N.S.** (2014). Real Time Learning of Rearranging Associations on a Humanoid Robot, *Bernstein Conference 2014.*, Göttingen, Germany.

[38] **Ercelik, E. and Sengor, N.S.** (2015). A neurocomputational model implemented on humanoid robot for learning action selection, *Neural Networks (IJCNN), 2015 International Joint Conference on*, IEEE, pp.1–6.

[39] **Ercelik, E.**, **Elibol, R. and Sengor, N.S.** (2015). A model on building and modifying the stimulus action association in the brain, *Signal Processing and Communications Applications Conference (SIU), 2015 23th*, IEEE, pp.2533–2536.

CURRICULUM VITAE

Name Surname : Emeç Erçelik

Place and Date of Birth : Denizli, 07.07.1990

Address : Istanbul Technical University, Faculty of Electrical and Electronics Engineering, Room:1117, Maslak, Istanbul

E-Mail : ercelike@itu.edu.tr

EDUCATION

B.Sc. : Istanbul Technical University (2013)
Electrical and Electronics Faculty
Electronics Engineering

M.Sc. : Istanbul Technical University (2016)
Graduate School of Science Engineering and Technology
Electronics Engineering

PUBLICATIONS/PRESENTATIONS ON THE THESIS

▪ **Erçelik, E.**, Denizdurduran B. and Şengör N. S.,2013: A Bioloid Robot Implementation of Computational Basal Ganglia Circuit. 11^{th} *National Neuroscience Congress*, Türkiye.

▪ **Erçelik, E.** and Şengör N. S.,2014: Implementation of Matching Stimulus-Movement Experiment on a Humanoid Robot. *International Workshop on Autonomous Cognitive Robotics*, Stirling, Scotland.

▪ **Erçelik, E.** and Şengör N. S.,2014: Uyaran-Eylem İlişkilendirme Ödevinin İnsansı Robot Üzerinde Gerçeklenmesi. 12^{th} *National Neuroscience Congress*, Türkiye.

▪ **Erçelik, E.** and Şengör N. S.,2014: Real Time Learning of Rearranging Associations on a Humanoid Robot. *Bernstein Conference 2014*, Göttingen, Germany.

▪ **Erçelik, E.** and Şengör N. S.,2015: A model on building and modifying the stimulus action association in the brain. *Signal Processing and Communications Applications Conference (SIU), 2015* 23^{th},pp.2533–2536.

▪ **Erçelik, E.**, Elibol R. and Şengör N. S.,2015: A neurocomputational model implemented on humanoid robot for learning action selection. *Neural Networks (IJCNN), 2015 International Joint Conference on*,pp.1-6.