Simulating Receptor Induced Cascades in Neurons – with Special Reference to Basal Ganglia

HUSAIN TALUKDAR

KTH Computer Science and Communication

Master of Science Thesis
Stockholm, Sweden 2011
Simulating Receptor Induced Cascades in Neurons – with Special Reference to Basal Ganglia

HUSAIN TALUKDAR

Master's Thesis in Biomedical Engineering (30 ECTS credits)
at the Computational and Systems Biology Master Programme
Royal Institute of Technology year 2011
Supervisor at CSC was Jeanette Hellgren Kotalaski
Examiner was Anders Lansner

TRITA-CSC-E 2011:078
ISRN-KTH/CSC/E--11/078--SE
ISSN-1653-5715

Royal Institute of Technology
School of Computer Science and Communication
KTH CSC
SE-100 44 Stockholm, Sweden
URL: www.kth.se/csc
ABSTRACT

In order to process and transmit information, chemical signaling occurs via synapses between neurons. This signaling uses a wide range of receptors, and synaptic information processing encompasses both biochemical reactions and molecule diffusions. One aim of this thesis is to understand receptor-induced cascades involved in synaptic plasticity in basal ganglia. To investigate the mechanism of signaling cascades, computational modeling of biochemical networks is essential. In particular, some aspects can be captured with deterministic models, but often modeling such cascades in a stochastic manner is necessary. In this thesis I have looked into ordinary differential equation (ODE) for deterministic simulation and NeuroRD a tool for stochastic reaction-diffusion simulation. I have investigated how NeuroRD is built from the structural and functional point of view to facilitate its use by others in the future.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

To the departed soul of my father-in-law ……

He died on the eve of starting this thesis work
ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to my supervisor Prof. Jeanette Hellgren Kotaleski for her support, enthusiasm and care. I am greatly moved by the way she motivated me through her insightful ideas, valuable inputs, and comments. I am deeply indebted to her for meticulously checking my work and for giving me her precious time even beyond the office hours.

I am thankful to Prof. Anders Lansner for giving me the opportunity to work in this project and Dr Michael Hanke, Mikael Djurfeldt, Shuoqi Zhang, Maya Brandi, Mohammed Samirul Karim for the enlightening discussions.

I would also like to thank my friends and family, especially my parents and in-laws. Although they are not directly involved in this thesis, they have provided me with valuable moral support and inspirations.

My deepest and utmost thanks and love goes to my wife, Tahmina Akhter, who assisted me in this thesis, directly by instructive discussions and indirectly by necessary distractions.
# Table of Contents

Chapter 1: Introduction ............................................................................................................. 1  
  1.1 Aim of the Thesis ................................................................................................................. 2  
  1.2 Scope of the Thesis ............................................................................................................... 2  
  1.3 Used tools and technologies ............................................................................................... 3  
Chapter 2: Biological Background ............................................................................................. 4  
  2.1 Basal Ganglia ....................................................................................................................... 4  
  2.2 The MSN (Medium Spiny Neurons) ..................................................................................... 6  
  2.3 Neurotransmitters and their Receptors ............................................................................... 7  
  2.4 Synaptic Plasticity ................................................................................................................ 8  
    2.4.1 LTP ................................................................................................................................. 8  
    2.4.2 LTD ............................................................................................................................... 9  
  2.5 Dopamine and Glutamate Receptor Cascades in MSNs ...................................................... 10  
Chapter 3: Models and Methods ............................................................................................... 11  
  3.1 Deterministic Models ......................................................................................................... 11  
  3.2 Stochastic Models ............................................................................................................... 13  
    3.2.1 Diffusion ....................................................................................................................... 13  
    3.2.2 Reactions ..................................................................................................................... 14  
Chapter 4: Results and Discussion ............................................................................................ 17  
  4.1 Deterministic Simulations ................................................................................................. 17  
  4.2 Stochastic Simulations ....................................................................................................... 25  
    4.2.1 Description of NeuroRD ......................................................................................... 25  
    4.2.2 Stochastic Simulation through NeuroRD ................................................................. 41  
  4.3 Discussion .......................................................................................................................... 44  
Chapter 5: Future Work .............................................................................................................. 46  
Bibliography ............................................................................................................................... 47  
Appendix A ............................................................................................................................... 50  
Appendix B ............................................................................................................................... 51  
Appendix C ............................................................................................................................... 54  
Appendix D ................................................................................................................................ 56
The List of Figures

Figure 1: Basal Ganglia in the forebrain region [14].................................................................4
Figure 2: Basal Ganglia Connectivity [16].................................................................................5
Figure 3: Medium spiny neuron [1:page420] .............................................................................6
Figure 4: Classification of Neurotransmitter receptors. The colored receptor play important role in the basal ganglia..........................................................7
Figure 5: LTP induction. A) D1 and A2A facilitate LTP in the direct and indirect pathway, respectively. B) A detailed view of the D1 / A2A receptor induced cascades..........................................................8
Figure 6: LTD induction..............................................................................................................9
Figure 7: The Lindskog model (left) and The Nakano model (right)..........................................10
Figure 8: Flowchart of diffusion in each time steps during simulation.......................................14
Figure 9: PKAc-PP2A-pThr75 loop of Lindskog model..............................................................17
Figure 10: Level of PKAc in the original Lindskog model when the loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]....................................................................................18
Figure 11: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]....................................................................................18
Figure 12: Level of PKAc in the original Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]....................................................................................19
Figure 13: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]....................................................................................20
Figure 14: Level of PKAc in the original Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]....................................................................................21
Figure 15: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]....................................................................................21
Figure 16: Level of PKAc in the original Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]....................................................................................22
Figure 17: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]....................................................................................23
Figure 18: Basic organizational flowchart of the NeuroRD.........................................................25
Figure 19: Dendrite compartments.............................................................................................27
Figure 20: Different parts of spine............................................................................................28
Figure 21: Typical 2D Neuron....................................................................................................30
Figure 22: Discretization of dendrite ........................................................................................33
Figure 23: Discretization of a spine..........................................................................................34
Figure 24: Typical 3D Neuron....................................................................................................35
Figure 25: Very small parts of Neuron (A) Dendrite (2x2x1), (B) Sub-membrane. (C) Spine ...........36
Figure 26: Diffusion steps in NeuroRD ....................................................................................37
Figure 27: Neighbors compartment where molecules can move............................................37
Figure 28: Reaction steps in NeuroRD.....................................................................................38
Figure 29: Output structure of NeuroRD ................................................................................40
Figure 30: Sample Graphical Output of NeuroRD using the tool NeuroRD_Output_Analyzer......40
Figure 31: Experiment 1, Species in dendrite (left) and the quantity table (right).......................41
Figure 32: Experiment 2 environment (left) and the result table (right).....................................41
Figure 33: Experiment 3 environment (left) and the result table (right).....................................42
Figure 34: Experiment 4 environment (left) and the result table (right).....................................42
Figure 35: Experiment 5 environment (left) and the result table (right).....................................43
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

Figure 36: Experiment 6 environment (left) and the result table (right).................................43
Figure 37: Working diagram of NeuroRD with source code reference.....................................51
Chapter 1: Introduction

Human is considered to be an advanced and developed specimen in the animal kingdom. In every situation, the human brain shows impressive abilities to discover structure in the environment and learn new things. Biologically, the brain is the source of intelligence and regulates how to perform actions. Indisputably, it is a challenging task to understand the complex structure and functional activities of the brain. Till now, the most advanced technology has not been able to fully unravel the mysterious brain and its function. Efforts are made throughout the world by the scientific communities to understand and discover the underlying workings of the brain.

Furthermore, all various aspects of the brain function, like perception, attention, cognition, emotions, rational thinking, behavior etc are organized and performed by a network called the human nervous system. In the human brain there are about 100 billion neurons and about \(10^{15}\) connections. Although neurons are separate nerve cells, they never function in isolation; they are organized into neural circuits that process specific information. The synaptic connection is a junction that permits a neuron to pass electrical or chemical signal to other neurons. Each nerve cell can receive 1-100,000 synaptic inputs. The communication between neurons is established by releasing chemical messengers called neurotransmitters. The large number of neurotransmitters and their receptors allow for tremendous diversity of chemical signaling between neurons.

According to the functionality, the human nervous system can be classified into three categories, Sensory systems (inputs), motor systems (outputs) and associational systems. Motor systems respond to information from sensory and associational systems by generating movements and other behaviors. From the anatomical point of view the basic parts of the brain are the brainstem (medulla, pons, midbrain), forebrain (diencephalon, cerebral hemispheres), and cerebellum. One of the structures to be considered here is the basal ganglia, which are situated in the cerebral hemisphere of the forebrain region and which are responsible for movement and decision making. Striatum is the major input stage of the basal ganglia and the MSN (Medium Spiny Neuron) [1] is the main cell population of the striatum.

The basal ganglia are not only important for controlling motor behavior, but also learning takes place here. Learning [2] is assumed to occur as a result of synaptic plasticity. Plasticity is controlled by neurotransmitter receptor-induced intracellular cascades. Disturbances in the basal ganglia functions, including the synaptic plasticity, result in diseases such as Parkinson’s disease, Huntington’s disease, as well as several neuropsychiatric disorders.

Scientists investigate the signaling network with the goal to understand the underlying mechanism of different brain functions. This approach also can lead to the discovery of the cause as well as to solutions of different brain diseases. Biological systems are inherently complex with numerous variables and their interactions over time. Thereby, it is difficult to perform, expensive to carry out, time consuming, and even some times impossible to perform experiments to unravel the brain function. In this case, computer modeling is an indispensable tool to simulate quantitatively the dynamic process going within between neurons. A model serves many purposes. It is a tool to investigate the role of every component of the signaling network and how they actually work together. The model should be simple and exact, while it should be efficient to simulate.
Computer simulation is a technique to visualize and realize a system before implementing the obtained model into the real world. To analyze and understand complex systems for example biochemical signaling network of the human brain, it is thus wise to use computer simulation.

It is quite fascinating to analyze and wonder about the complex functioning of the brain, for example the MSNs in the striatum. As is done below, I am particularly interested to know, in depth, about the signaling networks in the medium spiny projection neurons.

This thesis report is organized as follows:

In the second chapter, some biological backgrounds about the basal ganglia, MSN, neurotransmitter receptors, synaptic plasticity, and two models investigating intracellular signaling in MSN are described. In the third chapter, mathematical descriptions for both deterministic and stochastic modeling approaches are provided. The fourth chapter contains deterministic simulation results, descriptions of the NeuroRD simulator, and stochastic simulation results. Chapter five includes some directions of future works. Some findings of errors and mistakes in the NeuroRD tool are listed in Appendix A. Appendix B contain the reference of source code according to the NeuroRD structure. An abstract of ongoing future work is included in Appendix C, and Appendix D contains the input files of a toy model which has been used for investigating the stochastic NeuroRD simulator.

1.1 Aim of the Thesis

The aim of this thesis is to investigate the dopamine and glutamate receptor induced cascades in MSN. Deterministic and stochastic tool for simulating receptor induced signaling cascades in neuron are used. Specifically, I have in detail investigated a neuronal stochastic diffusion-reaction simulator, NeuroRD.

1.2 Scope of the Thesis

I study only the particular biochemical signaling network inside an MSN activated by dopamine and glutamate receptor. The MSN is the main neuronal population in striatum. The modeling work has two specific parts, the first one is deterministic modeling and the second one is stochastic modeling. To investigate the deterministic model I considered a feed-forward loop PKAc-PP2A-pThr75 from two previously published models, the Lindskog model [3] and Nakano model. This loop is a part of the dopamine and glutamate activated signaling network.

Before doing stochastic modeling I looked into a tool, NeuroRD. NeuroRD [2] is a software tool for simulating reaction diffusion systems taking into account the stochasticity of molecular interactions and diffusion. This tool merges Daniel T. Gillespie’s tau-leap algorithm [3] (part of Stochastic Simulation of Chemical Kinetics algorithm) with a stochastic diffusion algorithm [4].

Finally, by using the NeuroRD tool I investigate stochastic model. Here, I considered a neuronal signaling cascade which consists of membrane-bound species and free species.
1.3 Used tools and technologies

During the preparation of this thesis I used the following tools and technologies:

- MATLAB
- JAVA
- NeuroRD
- Microsoft Office Visio2010
- XML
Chapter 2: Biological Background

Below a general background is given for the basal ganglia, its input, outputs, and how synaptic plasticity can result from activation of receptor induced cascades.

2.1 Basal Ganglia

The Basal Ganglia are a group of nuclei in the brain of vertebrates, situated in the forebrain and receiving input from cerebral cortex, and thalamus.

![Figure 1: Basal Ganglia in the forebrain region](image)

The Basal Ganglia comprises three regions [11]:

i) Input region:
   Striatum is the input stage of basal ganglia. It includes dorsal striatum (caudate and putamen nucleus) and ventral striatum (nucleus accumbens). The main neuronal population of striatum is MSNs (Medium Spiny Neuron). About 96% of all cells are MSNs. Other populations are Cholinergic Neuron, and GABAergic parvalbumin, GABAergic calretinin, GABAergic somatostatin.

ii) Output region: The main output region, pallidum, comprises of Globus Pallidus Pars Interna (GPI) and Substantia Nigra Pars Reticulata (SNr). Output regions activate motor centers in thalamus and brainstem through disinhibition. Substantia Nigra Pars Compacta (SNc), where dopamine is produced lies in together Substantia Nigra Pars Reticulata (SNr). Dopamine activates metabotropic neurotransmitter receptors on mainly MSNs, which receive their main dopamine input from the substantia nigra pars compacta, and plays an important role in the coordination of the body movements[1:p147].

iii) Nuclei within Basal Ganglia:
   They consist of the subthalamic nucleus (STN) and Globus Pallidus Pars externa (GPe).
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

According to the classical view there are three pathways through the basal ganglia [2]:

- **Direct Pathway:**
  
  Cortex→Striatum→GPi/SNr
  
  Glutamate inputs from cortex drives the MSNs. Dopamine can facilitate the direct pathway through Striatonigral MSNs when it activates D1 receptors. These MSNs express high levels of D1 receptor.

- **Indirect Pathway:**
  
  Cortex→Striatum→GPe→Subthalamic Nucleus→GPi/SNr, Or
  
  Cortex→Striatum→GPe→GPi/SNr
  
  Dopamine can inhibit the indirect pathway through Striatopallidal MSNs. These MSNs express D2 receptor which gives the opposite behavior compared to D1 receptor. Also the indirect pathway is activated by glutamate.

- **Hyper direct Pathway:**
  
  Cortex→Subthalamic Nucleus→GPi/SNr

![Figure 2: Basal Ganglia Connectivity [16]](image)

[Red arrows indicate excitatory connections and blue arrows indicate inhibitory connections]
The majority of neurons in striatum are GABAergic medium spiny projection neurons which receive cortico-striatal inputs and project to globus pallidus. The subthalamic nucleus (STN) receives glutamatergic input from frontal cortex and cortical somato-motor areas [15]. Globus pallidus externa (GPe) receives input from striatum and STN and can project back to STN. Globus pallidus interna (GPi) receives GABAergic input from GPe and glutamatergic input from STN and projects both to thalamus and brain stem. Substantia nigra pars reticulata (SNr) and substantia nigra pars compacta (SNc) receive inputs from striatum [16]. Substantia nigra pars reticulata (SNr) also receive input from STN and inhibit thalamus.

The basal ganglia play a central role in a number of neurological conditions, including several movement disorders, like Parkinson’s disease (involves major loss of dopaminergic cells in Substantia Nigra Pars Compacta) [17] and Huntington’s disease (involves massive loss of MSN in striatum) [18].

Basal ganglia control motor behavior, and one well studied example is the contact of eye movements. Basal Ganglia have a permissive role for eye movement. Substantia Nigra pars reticulata projects to the upper motor neuron to act as a gate that must be opened to allow either sensory or other signal from cognitive centers to activate upper motor neurons and initiate eye movement such as saccades.

### 2.2 The MSN (Medium Spiny Neurons)

In striatum, about 96% of neurons are medium spiny projection neurons (MSN). Each MSN can receive tens of thousands of glutamatergic inputs from cortex and thalamus [2, 19].

![Medium spiny neuron](image)

**Figure 3: Medium spiny neuron [1:page420]**

According to the expression of dopamine receptor type D1 and D2 as well as the axonal projection site, two types of MSN’s are available in striatum [10, 13]:

1. **Striatopallidal MSNs**
   - MSN projects according to the indirect pathway.
   - Sends their principal axonal arbor to globus Pallidus externa.
   - Expresses high levels of D2.
2. Striatonigral MSNs
   - MSN projects according to the direct pathway.
   - Sends their principal axonal arbor to the Substantia Nigra / globus pallidus interna.
   - Expresses high levels of D1.

2.3 Neurotransmitters and their Receptors

Neurons in the human brain communicate with one another by releasing chemical messengers called neurotransmitters [1].

Neurotransmitters evoke postsynaptic electrical responses by binding to members of a diverse group of proteins called neurotransmitters receptors [1].

Two types of receptors are found in MSNs [1]:

- Ionotropic receptor or ligand-gated ion channels.  
  [Receptor molecules are also an ion channel]
- Metabotropic receptors.  
  [Receptor and ion channel are separate molecules]  
  Here the ion channel is opened or closed following the activation of a receptor induced cascades, e.g. leads to g-protein activation etc.

Various neurotransmitter receptors are depicted below. Several of those are found in the basal ganglia.

![Classification of Neurotransmitter receptors](image)

*Figure 4: Classification of Neurotransmitter receptors. The colored receptor play important role in the basal ganglia.*
2.4 Synaptic Plasticity

Synaptic plasticity [2, 10, and 31]: Synaptic plasticity is the ability to change the efficacy of synaptic connections. Synaptic transmissions can be either enhanced or depressed by activity for milliseconds to hours, days or even for a longer period of time. Depending on the duration of the activity of synaptic plasticity, it can be categorized in two types:

- Short term synaptic plasticity [2, 31]: Different forms of short term synaptic plasticity can last for a few milliseconds to several minutes.
- Long term synaptic plasticity.

Two principal form of Long term synaptic plasticity at glutamatergic synapses are [2]:

I. Long Term Potentiation (LTP)
II. Long Term Depression (LTD)

2.4.1 LTP

Striatal LTP is a long-lasting increase in the efficiency of glutamatergic synapses in the basal ganglia that is observed mainly at corticostriatal synapses [2, 20-21]. LTP predominates in dorsomedial and rostral striatum [2]. Different neurotransmitters are involved in LTP and it is partly known in what ways [2, 24-26].

Glutamate, dopamine and its receptors are critical for LTP induction and expression. Most forms of LTP observed in CNS require activation of NMDA-type glutamate receptors [2, 27-31]. AMPA-type glutamate receptors are also involved in the expression of striatal LTP [2].

LTP induction [2, 10, 24-31] for both the direct and indirect pathway is shown below,

Figure 5: LTP induction. A) D1 and A2A facilitate LTP in the direct and indirect pathway, respectively. B) A detailed view of the D1 / A2A receptor induced cascades.
2.4.2 LTD

Long-term reduction in synaptic efficacy also exists in glutamatergic synapse in striatum, which is known as long-term depression (LTD) [2, 10, 27, 31-34]. LTD requires [2, 10] high frequency afferent stimulation or timed pairing between activation pre- and post-synaptic neuronal elements (STDP) [30, 35-36].

Striatal LTD involves “retrograde signaling” which implies post-synaptic induction and pre-synaptic expression [37-39]. LTD induction and the different neurotransmitter receptors involved are shown below [10, 40-46]:

![Diagram of LTD induction](image)

*Figure 6: LTD induction*

Induction of LTD is post-synaptic and expression is pre-synaptic [10]. It is mainly dependent on L-type and mGluR, but NMDA dependent LTD is found in ventral striatum [10]. D2 facilitate indirect pathway in LTD and D1 facilitate direct pathway in LTP.
2.5 Dopamine and Glutamate Receptor Cascades in MSNs

Intracellular signaling networks consist of various signaling cascades in striatal MSNs. Dopamine and glutamate synaptic interactions are important for LTP and LTD. In this thesis, I considered two models of intracellular signaling network of dopamine and glutamate activated signaling cascades: the Lindskog model [3] and the Nakano model [13].

![BIOCHEMICAL NETWORK](image)

**Figure 7: The Lindskog model (left) and The Nakano model (right)**

The Lindskog and the Nakano model both try to understand how dopamine and glutamate produce plasticity.

The Lindskog and the Nakano model are biochemical signaling networks in Medium Spiny Projection Neuron (MSN) in striatum of Basal Ganglia. These models allows investigating the effect of transient and steady-state glutamate and dopamine input levels of signaling molecules and the phosphorylation of DARPP-32 on Thr34 and Thr75.

Both models have loop PKA-PP2A-pThr75 but in the Nakano model this loop behaves as a positive feed forward loop and in the Lindskog model it is described as a negative feed forward loop. In this thesis I have investigated the differences between two models.

These two models are quite similar when it comes to the PKA-PP2A-pThr75 loop except some differences, like in the Nakano model active PP2A cannot dephosphorylate DARPP32 at Thr34 and Thr75 but according to the Lindskog model active PP2A can dephosphorylate DARPP32. This was one difference we focused on.
Chapter 3: Models and Methods

To mathematically describe the time evolution of a well-stirred chemically reacting system, there are two formalisms that exist [47]:

- Deterministic approach
- Stochastic approach.

Usually, the state of well stirred chemically reacting system is defined by the current number of molecules or more often concentrations of each compartment species.

In the deterministic approach we can use ODE (Ordinary Differential Equation) to mathematically define the time evolution of chemical reactions from concentration of species and rate constants, and this is described in detail in the next section.

In the stochastic approach, the system state is considered by total number of molecules, but still ignores the positions and velocities of individual molecules in order to maintain the condition well-stirred. It is crucial to define the next reaction and the time of reaction in such system. Gillespie’s tau-leap algorithm [5, 7, and 49] is used to characterize these two constraints. Since, the stochastic simulations of biochemical network consist of diffusion in addition to reactions; a diffusion algorithm [6] is also necessary here.

3.1 Deterministic Models

A deterministic system is a system in which no randomness is involved in the future state of the system. A deterministic model always produces the same output from a given starting conditions or initial states.

Two types of reactions are involved in biochemical signaling networks [3, 13]:

I. Protein-Protein interactions

\[ A + B \rightleftharpoons AB \]

\( K_f \) and \( K_b \) are respectively, forward and backward rate constants.

II. Enzymatic reactions often described using the Michaelis-Menten form

\[ E + S \rightleftharpoons ES \rightarrow E + P \]

Where, E stands for Enzyme, S for Substrate and P for Products.

Equation I can be simulated by first order ODE (Ordinary Differential Equation) as follows:

\[ \frac{d[A]}{dt} = - K_f * A * B + K_b * AB \]

\[ \frac{d[B]}{dt} = - K_f * A * B + K_b * AB \]
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

\[ \frac{d[AB]}{dt} = K_f^*A*B - K_b^*AB \]

The following rules and formulas are also used in this deterministic model:

Dissociation constant, \( K_d = \frac{K_b}{K_f} \)

Affinity constant, \( K_a = \frac{K_f}{K_b} = \frac{1}{K_d} \)

Time constant, \( \tau = \frac{1}{K_f+K_b} \)

Therefore, \( K_f = \frac{1}{\tau*(1+K_d)} \) and \( K_b = \frac{K_d}{\tau*(1+K_d)} \)

Michaelis Constant, \( K_M = \frac{K_b+K_c}{K_f} \)

“Bhalla’s rule”, \( K_b \approx 4*K_c \) Therefore, \( K_f = \frac{5*K_c}{K_M} \)

\( V_{max} \) is the Maximum Enzyme Velocity.

From Michaelis-Menten we can derive,

\[ V_0 = \frac{V_{max}*[S]}{K_M+[S]} \]

where, \( V_0 \) is the initial reaction rate for a certain concentration of ‘S’.

Depending on total substrate concentration, [S] and Michaelis constant, \( K_M \) the rate of reaction \( \frac{dP}{dt} \) can be defined as follows:

i. If \([S] >> K_M\) then \( \frac{dP}{dt} \approx V_{max} = V_0 = K_c*[E]_0 \), where \([E]_0 \) stands for total enzyme.

ii. If \([S] = K_M\) then \( \frac{dP}{dt} = \frac{1}{2} * K_c*[E]_0 \) And \( K_M = \frac{1}{2} V_{max} \)

iii. If \([S] << K_M\) then \( \frac{dP}{dt} \approx \frac{V_{max}*[S]}{K_M} = \frac{K_c}{K_M}*[E]_0*[S] \)

N.B If enzyme concentration is high, then the production of product will be high.
3.2 Stochastic Models

Stochastic modeling [1] is a technique for estimating the potential outcomes by allowing random variations in one or more inputs over time. It is noted that, in a biochemical network system, diffusion and reactions occur together. Therefore, both diffusion and reaction models are necessary to describe here.

In a biochemical reacting system, the state of the system cannot progress with time as a deterministic system. Due to small number of molecules the outcome is different between trials. In addition, some molecules are stationary and some are floating around. By compartmentalization (divides neuron into small compartment) it is easy to define and understand the movement and collision between molecules.

3.2.1 Diffusion

In NeuroRD (Stochastic Neuronal Reaction diffusion simulator), a stochastic diffusion algorithm [6] has been implemented. The probability of a molecule to leave a spatially discrete compartment is used to create a lookup table that stores the probability of molecules leaving the compartment as a function of the total number of molecules in the compartment. During simulation, the total number of molecules leaving the compartment is determined using a uniform random number as an index into the lookup table.

This diffusion algorithm has two parts:

I. Initialization: the following steps have been done during initialization,
   - Subdividing the dendrite into equal size compartments.
   - Choosing time step.
   - Creating a connection array to identify the neighbors of each compartment.
   - Calculating the probability of moving molecules to adjacent compartment as follows,
     \[ P_m = 2D \frac{\Delta T}{\Delta x^2} \]
     where, \( P_m = \) probability of moving molecules
     \( D = \) diffusion coefficient
     \( \Delta T = \) time step; \( \Delta x = \) length of the compartment
   - Creating the table of probabilities, \( km \) out of \( N \) molecules, that leaves the compartment, for \( N \) between 1 and \( N_{\text{max}} \) (100) using binomial distribution.
   - Creating another table to store the probability of moving direction

II. Simulation: During each simulation, the steps are described through the following flowchart:
3.2.2 Reactions

The details of the time evolution of a well-stirred chemically reacting system, e.g. in MSNs are shown here using the Gillespie’s tau-leap [5, 7] algorithm:

Let us consider a well-stirred system of molecules of, N chemical species \(\{S_1, \ldots, S_N\}\), M chemical reactions \(\{R_1, \ldots, R_M\}\), total volume and temperature is constant.

\(X_i(t)\) denotes the number of molecules of species \(S_i\) in the system at time \(t\).

Our goal is to calculate, \(X(t) \equiv (X_1(t), \ldots, X_N(t))\)

where, state \(X(t_0) = x_0\) at initial time \(t_0\)

The change of the state according to reactions can be characterized mathematically by two quantities; the state-change vector and the propensity function.

**State-change vector**

\(V_j \equiv (V_{1j}, \ldots, V_{Nj})\), where

\(V_{ij} = \text{change of the } S_i \text{ molecular population caused by } R_j \text{ reaction.}\)

So, if current state \(x\) and \(R_j\) occur then immediately it will jump to state \(x + V_j\)
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

**Propensity function**

\( a_j(x) \, dt \triangleq \) the probability, given \( X(t) = x \), that one \( R_j \) reaction will occur somewhere inside \( \Omega \) in the next infinitesimal time interval \([t, t+dt]\).

Depending on reaction it will be different [],.

For unimolecular reaction \( A \leftrightarrow B \)

\[ a_j(x) = k_j \cdot x_1 \] where \( k_j \) is rate constant

For bimolecular with two different species, \( A + B \rightarrow C \)

\[ a_j(x) = \frac{k_j}{\Omega} \cdot x_1 \cdot x_2 \]

For bimolecular with same species, \( A + A \rightarrow B \)

\[ a_j(x) = \frac{2k_j}{\Omega} \cdot \frac{x_1(x_1-1)}{2} \]

This probability can be written as follows,

\[ P(x, t \mid x_o, t_0) \triangleq \text{Prob} \{ X(t) = x, \text{given} X(t_0) = x_0 \} \]

And we can derive a time-evolution equation for this probability using the chemical master equation (CME) [48], as follows:

\[
\frac{\partial P(x, t \mid x_o, t_0)}{\partial t} = \sum_{j=1}^{M} \left[ a_j(x - v_j)P(x - v_j, t \mid x_o, t_0) - a_j(x)P(x, t \mid x_o, t_0) \right]
\]

Using CME, we can calculate the probability of the next state of the system, but we don’t know the next reaction and the reaction time. To find this out, we introduce a new probability function:

\[ p(\tau, j \mid x, t) \, d\tau \triangleq \text{the probability, given} \ X(t) = x, \text{that the next reaction in the system will occur in the infinitesimal time interval} \ [t + \tau, t + \tau + d\tau], \text{and will be an} \ R_j \text{reaction.} \]

This probability can be derived as follows:

\[ p(\tau, j \mid x, t) = a_j(x) \exp(- a_0(x) \, \tau), \]

Where, \( a_0(x) \triangleq \sum_{j=1}^{M} a_j(x) \).

This is the basis of the stochastic simulation approach.

Now the question is, how can we calculate the value of \( \tau \) and \( j \) (when and which reaction)? In the Stochastic simulation algorithm (SSA) we use the direct method [51] by drawing two random numbers \( r_1 \) and \( r_2 \) from the uniform distribution in the unit interval. Therefore, we get,

\[ \tau = \frac{1}{a_0(x)} \ln \left( \frac{1}{r_1} \right), \]

\[ j = \text{the smallest integer satisfying} \sum_{j=1}^{j} a_j(x) > r_2 \cdot a_0(x) \]
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

According to this, the SSA [7, 47, and 51] steps are:

1. Initializing the time $t = t_0$ and the system’s state $x = x_0$.
2. With the system in state $x$ at time $t$, evaluating all the $a_j(x)$ and their sum $a_0(x)$.
3. Generating values for $\tau$ and $j$ using above equations.
4. Affecting the reaction by replacing $t \leftarrow t + \tau$ and $x \leftarrow x + v_j$.
5. Recording $(x, t)$ and returning step 1 or ending the simulation.

This SSA is exact but it simulates every reaction event one at a time and thereby often it takes too much time to implement for real systems [49].

**Tau-Leaping**

Tau-leaping is a way of accelerating the SSA in which each time step $\tau$ advances the system through possibly many reactions events. Leap-condition assures that no propensity function changes significantly during a small time $\tau$ [7].

The basic tau-leaping formula [49] is shown below,

$$X (t + \tau) = x + \sum_{j=1}^{M} v_j P_j (a_j(x), \tau)$$

Where, $P_j (a_j(x), \tau)$ for each $j = 1, \ldots, M$ denotes an independent sample of the Poisson random variable with mean $a_j(x), \tau$.

If $a_j(x), \tau >> 1$ for all $j = 1, \ldots, M$, it is easy to reduce the above basic equation by using the simple Euler method for the chemical Langevin equation (CLE) [52].

In NeuroRD fixed time steps have been used and that’s why the tau selection procedure is not described here.

The complete derivation of tau-leaping is described in Gillespie’s paper “Efficient step size selection for the tau-leaping simulation method” [49].
Chapter 4: Results and Discussion

Here, results are arranged in three distinct sections. The first part contains deterministic experiments, the investigation of a feed-forward loop in the Lindskog model and the Nakano model. The second part contains the structural and functional description of NeuroRD, a neuronal stochastic reaction diffusion simulator and the third part contains stochastic experiments, the investigation of biochemical cascades when consists of membrane bound species and free species. The aim of part two and three is to prepare for the final implementation of the Lindskog model using NeuroRD.

4.1 Deterministic Simulations

Our aim is to investigate the feed forward loop PKAc-PP2A-pThr75 according to the Lindskog model and the same loop in the Nakano model. First we have done a deterministic simulation according to the Lindskog model and then we adjusted the Lindskog model to mimic the Nakano model with regard to this loop. We have found from comparison between two models that, PP2A in the ground state cannot dephosphorylate DARPP-32 at Thr34 and Thr75 in the Nakano model, but in the Lindskog model PP2A can dephosphorylate DARPP-32 at Thr34 and Thr75. We therefore adjusted the Lindskog model by setting rate constant at 0 for the equation between PP2A and DARPP32 at Thr34 and Thr75, and by putting the initial value at 1200 for cdk5 instead of 1800. This was done to keep the same initial level of phosphorylated DARPP-32.

During the investigation we used dopamine (transient and steady-state) input with different amplitude and calcium input (at base level).

Every experiment has been done for both the intact loop and the blocked loop for the Lindskog model as well as for mimicking the Nakano model (by adjusting the Lindskog model). All reactions, rate constants and initial conditions have been collected from Lindskog model [3].

For the Lindskog model, every simulation has been started from steady state for 2000 ms and dopamine injection started after 500 ms (millisecond). And for the adjusted Lindskog model (mimicking Nakano model), every simulation has been started from initial conditions set by 0 or at base level for 7000 ms where after 4000 ms the feed forward loop, either in the intact or blocked mode and after 5000 ms dopamine injection started. Unit of species was nM (nano Mole). The result of the simulations is presented here.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

**Transient Input:**

Following two experiments have been done by using Dopamine (DA) transient input at 1000 nM and 20000 nM amplitude level. The original Lindskog model starts from steady state and run for 2000 second where injecting at 500s. And the adjusted Lindskog model starts from initial state and run for 7000 second where at 2000s it was on steady state and injecting at 5000s.

**Experiment 1:** Has been done using Dopamine (DA) transient input with 1000 nM amplitude level.

![Figure 10: Level of PKAc in the original Lindskog model when the loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]](image)

![Figure 11: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]](image)
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

Table 1: Output for dopamine transient input using 1000 nM amplitude according to the original Lindskog model.

<table>
<thead>
<tr>
<th>Transient Input, When loop is intact (Max amplitude 1000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>6.557</td>
<td>1404</td>
<td>36700</td>
<td>798.9</td>
<td>23.56</td>
<td>9603</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transient Input, When loop is Block (Max amplitude 1000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>7.772</td>
<td>1417</td>
<td>33400</td>
<td>1108</td>
<td>12.4</td>
<td>12900</td>
</tr>
</tbody>
</table>

Table 2: Output for dopamine transient input using 1000 nM amplitude according to the adjusted Lindskog model.

<table>
<thead>
<tr>
<th>Transient Input, When loop is intact (Max amplitude 1000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>6.517</td>
<td>2108</td>
<td>39030</td>
<td>886.7</td>
<td>17.55</td>
<td>7278</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transient Input, When loop is Block (Max amplitude 1000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>7.667</td>
<td>2190</td>
<td>33900</td>
<td>1285</td>
<td>11.96</td>
<td>12350</td>
</tr>
</tbody>
</table>

**Experiment 2:** Has been done using Dopamine transient input at 20000 nM amplitude level (i.e. very high DA level)

![Figure 12: Level of PKAc in the original Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]](image)
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

Figure 13: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]

Table 3: Output for dopamine transient input using 20000 nM amplitudes according to the original Lindskog model.

<table>
<thead>
<tr>
<th>Transient Input, When loop is intact (Max amplitude 20000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>11.05</td>
<td>3406</td>
<td>35700</td>
<td>584.8</td>
<td>35.64</td>
<td>8443</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transient Input, When loop is Block (Max amplitude 20000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>13.55</td>
<td>3045</td>
<td>31410</td>
<td>1050</td>
<td>12.4</td>
<td>13090</td>
</tr>
</tbody>
</table>

Table 4: Output for dopamine transient input using 20000 nM amplitudes according to the adjusted Lindskog model.

<table>
<thead>
<tr>
<th>Transient Input, When loop is intact (Max amplitude 20000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>11.26</td>
<td>6586</td>
<td>35900</td>
<td>647.2</td>
<td>23.87</td>
<td>5727</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transient Input, When loop is Block (Max amplitude 20000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>13.83</td>
<td>6790</td>
<td>29470</td>
<td>1285</td>
<td>11.96</td>
<td>12010</td>
</tr>
</tbody>
</table>

Using transient inputs PKAc is always larger if the loop is blocked. However, sometimes the pThr34 can be higher if the loop is intact and if the dopamine input is high in the adjusted Lindskog model.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

**Steady-state Input:**

To understand the result from the above simulation, we have again done the experiment 1 and 2 but using steady-state input to test the result depends on dynamics or not.

**Experiment 3:** Has been done using Dopamine (DA) steady-state input at 1000 nM amplitude level.

![Figure 14](image1.png)

*Figure 14: Level of PKAc in the original Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]*

![Figure 15](image2.png)

*Figure 15: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 1000 nM]*
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

**Table 5:** Output for dopamine steady-state input using 1000 nM amplitudes according to the original Lindskog model.

<table>
<thead>
<tr>
<th>Species</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>intact</td>
<td>20.1</td>
<td>10000</td>
<td>29390</td>
<td>369.7</td>
<td>59</td>
<td>7548</td>
</tr>
<tr>
<td>blocked</td>
<td>25</td>
<td>7220</td>
<td>26700</td>
<td>927.6</td>
<td>12.42</td>
<td>13360</td>
</tr>
</tbody>
</table>

**Table 6:** Output for dopamine steady-state input using 1000 nM amplitudes according to the adjusted Lindskog model.

<table>
<thead>
<tr>
<th>Species</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>intact</td>
<td>23.02</td>
<td>22210</td>
<td>21690</td>
<td>388.3</td>
<td>36.41</td>
<td>4275</td>
</tr>
<tr>
<td>blocked</td>
<td>29.95</td>
<td>21330</td>
<td>16600</td>
<td>1285</td>
<td>11.96</td>
<td>10330</td>
</tr>
</tbody>
</table>

**Experiment 4:** Has been done using dopamine (DA) steady-state input at 20000 nM amplitude level (i.e. very high DA level)

![Graph of PKAc in the original Lindskog model](image1)

**Figure 16:** Level of PKAc in the original Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

Figure 17: Level of PKAc in the adjusted Lindskog model when loop intact (upper) and blocked (lower) [DA amplitude 20000 nM]

Table 7: Output for dopamine steady-state input using 20000 nM amplitudes according to the original Lindskog model.

<table>
<thead>
<tr>
<th>Steady-state Input, When loop is intact (Max amplitude 20000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>34.56</td>
<td>21070</td>
<td>19690</td>
<td>235.6</td>
<td>92.61</td>
<td>6593</td>
</tr>
</tbody>
</table>

Table 8: Output for dopamine steady-state input using 20000 nM amplitudes according to the adjusted Lindskog model.

<table>
<thead>
<tr>
<th>Steady-state Input, When loop is intact (Max amplitude 20000)</th>
<th>PKAc</th>
<th>pThr34</th>
<th>DARPP32</th>
<th>PP2A</th>
<th>PKAi</th>
<th>pThr75</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKAc</td>
<td>42.42</td>
<td>32830</td>
<td>12170</td>
<td>236</td>
<td>51.67</td>
<td>3292</td>
</tr>
</tbody>
</table>

With a steady-state dopamine input, it turns out that pThr34 always is higher if the loop is intact, and despite the fact that PKAc is always larger when the loop is blocked. This is explained by the observations that activated PP2A can dephosphorylate pThr75 and thus shift the balance pThr75 ↔ DARPP32 ↔ pThr34. Thus pThr34 can be increased due to the lowering of the pThr75 despite less activated pThr34.
Final conclusion:

- PKAc always small when loop is intact, which means the PKAc-PP2A-pThr75 feed-forward loop behaving as a negative feed-forward loop.
- pThr34 either follow PKAc or, if dopamine input is large or has a long duration, the pThr34 can be higher if loop is intact.
- During dopamine transient input, only pThr34 depends on the dopamine amplitude within the original Lindskog model. Here, amplitude is proportional to pThr34 when loop is intact. But with the adjusted Lindskog model pThr34 is always small when loop is intact.
- During dopamine steady-state input, only DARPP32 depends on the dopamine amplitude within the original Lindskog model. Here, amplitude is reversely proportional to DARPP32 when loop is intact. But with the adjusted Lindskog model DARPP32 is always large when loop is intact.
4.2 Stochastic Simulations

The future goal is to implement and extend the Lindskog model using a stochastic approach. Thereby an important part of the diploma work was to test NeuroRD. In NeuroRD, the biochemical system state is described by total number of molecules where diffusion and reactions occur together. But we do not know when and how molecules will diffuse and how many will react. To compare that it is assumed neurons are virtually divided into a number of compartments. In this thesis for stochastic simulation I used a novel simulator called NeuroRD, which I describe in terms of structural and functional properties.

4.2.1 Description of NeuroRD

NeuroRD is a novel simulator for stochastic simulation of biochemical signaling network. NeuroRD stands for Neuronal Reaction Diffusion. The basic organizational flowchart of NeuroRD is shown below.

![Figure 18: Basic organizational flowchart of the NeuroRD.](image)
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

This flow chart is based on going through the NeuroRD source code. According to this flow chart, the functionality of every step is described below. Everything is exemplified with components corresponding to the Lindskog model in a dendrite with a spine connected to it and with realistic Ca dynamics.

**Input Files**

To run NeuroRD simulator, six input files are required. All files should be in XML format.

I. Master or model file  
II. Morphology  
III. Reaction scheme  
IV. Initial condition  
V. Output scheme file  
VI. Stimulation file  

I. Master File or model File

The model file contains the information of the other input files which provide information regarding to simulation, and some other basic information of the system. Tags (commands) of this model XML file are described below.

Tags to define the file names:

- `<reactionSchemeFile>MSPNreactions_new_1spine</reactionSchemeFile>`  
- `<morphologyFile>MSPNmorph_1spinea</morphologyFile>`  
- `<stimulationFile>MSPNstim_5DaStim</stimulationFile>`  
- `<initialConditionsFile>MSPNic_A</initialConditionsFile>`  
- `<outputSchemeFile>MSPNio_1spine_newmorph</outputSchemeFile>`

Tags about other basic information of the system:

```xml
//--------------------------Neuron discretization--------------------------
• `<MaxElementSide region="dend1">0.12</MaxElementSide>`  
  //---maximum length size for specific dendrite discretization---  
• `<defaultMaxElementSide>0.2</defaultMaxElementSide>`  
  //---default maximum length size for dendrite discretization---  
• `<spineDeltaX>0.1</spineDeltaX>`  
  //---maximum length size for spine discretization---  
• `<geometry>2D</geometry>`  
  //---neuron structure its can be either 2 dimensional or 3 dimensional---  
• `<depth2D>0.4</depth2D>`  
  //---mention the depth of dendrite---
```

```xml
//-----------------------------Algorithm-----------------------------
• `<distribution>BINOMIAL</distribution>`  
  //---define the algorithm of distribution like Binomoial or Poission---  
• `<algorithm>INDEPENDENT</algorithm>`  
  //---specify the diffusion algorithm like Independent or Shared or Prticle---  
• `<calculation>GRID_STEPED_STOCHASTIC</calculation>`
```
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

//---Specify the reaction algorithm like grid_stepped_stochastic or grid_stepped_continuous---

//-------------Define Time for simulation-----------------------------
- <runtime>400000</runtime> //--Define time step for simulation--
- <fixedStepDt>0.005</fixedStepDt> //--Define time step for simulation--
- <outputInterval>3000000.0</outputInterval> //---write the output from buffer to output file after this time---

//------------Seed Value for random function [Merchantwister Random Function]-------------
- <simulationSeed>000123</simulationSeed> //---Seed value for random function---
- <spineSeed>123</spineSeed> //---Seed value for random function to count the total number of spine----
- <tolerance>0.001</tolerance> //---is not used in the simulation---
- <outputQuantity>NUMBER</outputQuantity> //---specify the output molecules for every species as a number of molecules or the concentration [CONCENTRATION]

N.B
i. The property <tolerance> is not used in this simulation
ii. The property <outputQuantity> also has no effect in this simulator. It’s always show the output as a concentration of species.

H. Morphology

Morphology file contains the information regarding geometrical structure of each and every dendrite and spine. It is also defining the injecting point for stimulation during simulation.

Dendrite: If you have three connecting dendritic compartments like the following.

![Dendrite compartments](image)

*Figure 19: Dendrite compartments*

Then you have to prepare an input file as follows:

```xml
<Segment id="seg1" region="dend1">
  <start x="1.0" y="1.0" z="0.0" r="0.3" />
  <end x="3.5" y="1.0" z="0.0" r="0.3" />
</Segment>
<Segment id="seg2" region="dend2">
  <start on="seg1" at="end" />
```
If you want to inject molecules in any dendritic compartment you have to label the dendrite. The rules for labeling dendrite for injection are given below:

- Each segment/dendrite has two points, starting point and ending point. You have to label the starting point of the dendrite to inject something.

- If the starting point is a connecting point with the previous one, then you have to assign a label in the end point of the previous dendrite/segment (which is also the start point of the current dendrite). This is because you cannot define a label on a connecting point.

- In the case of the last dendrite/segment, you can define a label in either point; starting or ending point.

According to above statement the injection will affect the dendrite3, although labeling is shown on the end point of dendrite2. This is because the starting point of dendrite3 is a connecting point with dendrite2.

**Spine:** If we consider a spine as follows, which is defined in a different way, only width is used.

![Figure 20: Different parts of spine.](image)

Then, you have to define the geometrical structure of spine as follows:
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

```xml
<SpineType id="spineA">
  <Section width="0.2" at="0.0" regionClass="neck" />
  <Section width="0.2" at="0.2" regionClass="neck" />
  <Section width="0.2" at="0.3" regionClass="head" />
  <Section width="0.6" at="0.4" regionClass="head" />
  <Section width="0.6" at="0.5" regionClass="PSD" />
  <Section width="0.2" at="0.6" label="pointA" />
</SpineType>

Usually, in NeuroRD spine is considered as a frustum.

Now the question is, with which dendrite will the spine connect and how many spines will connect with this particular dendrite? To specify this, we have to follow certain rules:

```xml
<SpineAllocation id="sa1" spineType="spineA" region="dend3" lengthDensity="0.4" />
```

This means, we want to add “spineA” type spine on dendrite3. And the number of spines will be defined by lengthDensity or areaDensity.

lengthDensity means the average number of spines per micron of dendrite length.

areaDensity means the average number of spines per unit area.

**N.B:** In NeuroRD we always calculate the number of spines according to areaDensity.

If you want to inject into the spine then,

- You have to label the starting point of the region of the spine.
- In case of the last region, you can label either the starting point or the ending point.

In the given sample code, injection will be affected on PSD, although PSD is the last region. That is why it is labeled on the end point. You can also label the starting point of PSD.

According to the above description, the neuron looks as follows:
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

III. Reaction Scheme

The reaction scheme has two parts:

In the first part you have to assign every species’ id, name, diffusion constant and unit as follows:

```xml
<Specie name="Ca" id="Ca" kdiff="174.3" kdiffunit="mu2/s" />
```

And in the second part, you have to assign every reaction with forward and backward rate constants as follows:

```xml
<Reaction name="Ca_pump1a" id="Ca_pump1a">
  <Reactant specieID="Ca" />
  <Reactant specieID="pmca" />
  <Product specieID="pmcaCa" />
  <forwardRate>0.5e-4</forwardRate>
  <reverseRate>0.007</reverseRate>
  <Q10>0.2</Q10> //---this property is yet not used in this simulator---/
</Reaction>
```

Exception one,

```xml
<Reaction name="CamCa2+2Ca--CamCa4_id" id="CamCa2+2Ca--CamCa4_id">
  <Reactant specieID="CamCa2" />
  <Reactant specieID="Ca" n="2" />
  <Product specieID="CamCa4" />
  <forwardRate>0.1e-03</forwardRate>
  <reverseRate>1000e-03</reverseRate>
  <Q10>0.2</Q10>
</Reaction>
```

Here is an extra property ‘n’ used to define how many numbers of molecules of respective species participate in the reaction. By default n=1, or else you will have to set the value.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

IV. Initial Condition

The initial condition file specifies the initial concentration or densities for all molecular species. Two types of molecular species are found in dendrite, membrane-bound species and free species. For free species you have to set the concentration and for membrane-bound species you have to set the density of species. Species in spine tracked separately in concentration due to one dimensional diffusion.

The file can be divided in three parts:

- The first part is mandatory and it specifies a general concentration of all species, unless overridden. The unit of concentration is nanomoles per litre.
  
  <ConcentrationSet>
  <NanoMolarity specieID="Ca" value="58.669" />
  <NanoMolarity specieID="CaOut" value="2001468.5" />
  
  .
  
  .
  
  .
  </ConcentrationSet>

- The second part specifies the density of species in the sub-membrane region. Its unit is picomoles per square meter.
  
  <SurfaceDensitySet region="dend1">
  <PicoSD specieID="pmca" value="86.82" />
  <PicoSD specieID="pmcaCa" value="28.25" />
  <PicoSD specieID="ncx" value="2994.17" />
  </SurfaceDensitySet>

If you can’t specify the region then it will be applicable for all sub-membrane regions.

- The third part contains the concentration of species for a specific region of spine. It is also in nanomoles per liter.
  
  <ConcentrationSet region="neck">
  <NanoMolarity specieID="pmca" value="297.765" />
  <NanoMolarity specieID="pmcaCa" value="96.909" />
  <NanoMolarity specieID="ncx" value="10268.46" />
  </ConcentrationSet>

V. Output scheme file

In the output scheme file, you can define any number of output files from where you can get your desired outputs. You can set the output file for specific species, with respect to specific region, and with a time interval at which the rate is written out.

  <OutputSet region="PSD" filename="cell1" dt="10.0">
  <OutputSpecie name="GluR1" />
  <OutputSpecie name="GluR1_S845" />
  <OutputSpecie name="GluR1_S831" />
  
  .
  
  .
  
  .
  </OutputSet>

If you omit the region, then the output will produce results for all regions.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

If you omit the dt, then the output will produce results for every time step.

By default, NeuroRD produces two output files with your specified output file. These two output files are:

- mesh.txt file, which contains co-ordinates of all sub compartments with volume and deltax.
- Model.out, which contains co-ordinates of all sub-compartments and concentration of all species in all sub-compartments.

VI. Stimulation file

Note that input signals to the model occur through injection or in some cases, using initial condition values. For example a dopamine ligand, here L is injected at a certain time, Ca coming into cell, transmitter binding to receptor etc.

The stimulation file specifies the time and location of injection of molecules during the simulation.

```
<InjectionStim specieID="L" injectionSite="pointA">
  <onset>50000</onset> //---Injection time----
  <duration>5</duration> //----duration (msec) of inject---
  <rate>8</rate> //-----particles per msec-----
  //-------these are extra for train of input-------
  <period>20</period>
  <end>51000</end>
</InjectionStim>
```

You can also inject on a specific spine,

```
<InjectionStim specieID="L" injectionSite="sa1[0].pointA">
  <onset>70000</onset>
  <duration>5</duration>
  <rate>8</rate>
  <period>20</period>
  <end>71000</end>
</InjectionStim>
```

Where sa1 is the spine allocation id which is defined in morphology file and index number is the index of the spine. You can specify more spines by,

- using comma [2,3,5]
- all spine using [*]
- specific spine range using [1:4]
Discretization

**Dendrite discretization:** The dendrite is discretized two-dimensionally as shown below:

![Discretization of dendrite](image)

To discretize (sub divide) a dendrite, you have to calculate the maximum length of sub volumes using three parameters, defaultMaxElementSide d, Specific dendrite MaxElementSide and radius, r as per the following algorithm:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Take defaultMaxElementSide, d</td>
</tr>
</tbody>
</table>
| 2    | Calculate Candidate Grid Size for all end points of dendrite  
     |     | [Iterate through all end points with their radius.  
     |     | Divide each diameter by successively increasing odd numbers until the divided  
     |     | value becomes less than defaultMaxElementSide, d]  
| 2(a) | diameter = r*2.0; denominator = 1;  
     |     | While ( (diameter/denominator) > d)  
     |     | denominator+=2.0  
     |     | End While  
     |     | Candidate_Grid_Size [] = diameter/denominator; |
| 3    | Find out Min Value between d and all Candidate Grid Sizes. |
| 4    | Take Individual Max ElementSide |
| 5    | If Step4 is not exist then Step3 gives the grid size else Step4 gives the grid size |
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

**Spine discretization:** Spine discretized one-dimensionally according to length of sub volume (spineDeltaX) is shown below:

![Discretization of a spine](image)

*Figure 23: Discretization of a spine*

Calculation of the total number of spines from lengthDensity is shown below:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Take the density of spine</td>
</tr>
<tr>
<td>2</td>
<td>Calculate the total area of respective region (dendrite)</td>
</tr>
<tr>
<td>3</td>
<td>Calculate Average no. of spine = area*density</td>
</tr>
</tbody>
</table>
| 4    | if (avgnospine < 12)  
  g = Math.exp(-avgnospine)  
  t=1.0; em= -1;  
  do  
  em += 1.0;  
  t *= random();  
  while(t > g) |
| 5    | em is the total number of spine |
Initialization

During initialization, the following tasks have been done according to the input files information,

1) Calculating the number of molecules for each sub-compartment.
2) Determine the probability distribution (either binomial or Poisson) of molecules into each sub-compartment.
3) Create a connection array of the sub-compartments in neuron to identify the neighbor list of each sub-compartment.
4) Calculate the probability of a molecule moving to the neighboring compartments.

Calculate number of molecules:

Molecules can be spread in the neuron either two-dimensionally or three-dimensionally, depending on the region and type of species. Therefore, the number of molecules in a neuron can be calculated by considering three different positions and type of species:

- Free species in dendrite
- Membrane-bound species in dendrite
- Free species in spine
Before calculating the number of molecules we have to know the method to calculate:

1. Volume of dendrite (Fig. (A)),
   \[ \text{Volume} = \text{length} \times \text{width} \times \text{depth} \]

2. Area of membrane bound compartment of dendrite (Fig. (B)),
   \[ \text{Area} = \text{length} \times \text{depth} \]

3. Volume of spine (Fig. (C)), [spine is considered as a frustum]
   \[ \text{Volume} = \frac{\pi h (r^2 + rR + R^2)}{3} \]

And, total number of molecules of a free species in a sub-compartment is,

\[ \text{Volume of sub-compartment of dendrite} \times \text{Concentration of species} \times \text{NA}^1 \]

Total number of molecules of a membrane bound species in a sub-compartment is,

\[ \text{Area} \times \text{Density of species} \times \text{NA} \]

Total number of molecule of a species in a segment of spine is,

\[ \text{Volume of spine segment} \times \text{Concentration of species} \times \text{NA} \]

**Stimulation**

At every time step, NeuroRD checks the stimulation table as follows:

- Whether any injection exists or not at this time step.
- If any injection exists, then calculation of the total injecting number of molecules is done.
- Updating this into respective compartments according to species.

---

1 NA = Avogadro’s Number 6.022e23
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

**Diffusion**

The work flow of diffusion in NeuroRD is shown below. There are three algorithms; Independent, Shared and Particle that exist in NeuroRD. Only Independent algorithm has been tested in this thesis work. Blackwell’s diffusion algorithm has been implemented in NeuroRD, but not identical to publish paper. [See Appendix A]

![Diffusion Diagram](image)

**Figure 26: Diffusion steps in NeuroRD**

According to the NeuroRD every compartment has maximum four neighbors, though according to Blackwell’s diffusion algorithm one compartment can have at best eight neighbors.

![Diagram](image)

**Figure 27: Neighbors compartment where molecules can move**
Reactions

The change of states (number of molecules of species) in the biochemical network system has been done according to this flow.

Gillespie’s tau-leap algorithm has been implemented in NeuroRD.

![Figure 28: Reaction steps in NeuroRD](image)

Leap condition checked by calculating probability of event for one particle, \( p \) as follows:

- For bimolecular reaction (different species)
  \[
  p = \frac{\text{rate of constant} \times \text{maximum number of molecules between two species} \times \text{time step}}{\text{volume} \times \text{NA}}
  \]

- For bimolecular reaction (same species)
  \[
  p = \frac{\text{rate of constant} \times \text{number of molecules} \times \text{time step}}{\text{volume} \times \text{NA}}
  \]

- For unimolecular reaction
  \[
  p = \text{rate of constant} \times \text{time step}
  \]

After each reaction, total number of going molecules, ‘ngo’ is calculated according to Gillespie by using either Binomial or Poisson distribution.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

In every time step the number molecules for each species has been updated according to the following steps:

- $r_i0 = \text{number of molecules of first reactant}$
- $r_i1 = \text{number of molecules of second reactant}$
- $r_s0 = \text{Stoichiometric value of first reactant}$
- $r_s1 = \text{Stoichiometric value of second reactant}$
- $p_i0 = \text{number of molecules of first product}$
- $p_i1 = \text{number of molecules of second product}$
- $p_s0 = \text{Stoichiometric value of first product}$
- $p_s1 = \text{Stoichiometric value of second product}$

**Step1:** Calculate number of available molecules for each species, ‘navail’

- For bimolecular reaction, $\text{navail} = \min(r_i0/r_s0, r_i1/r_s1)$
- For unimolecular reaction, $\text{navail} = r_i0/r_s0$

**Step2:** Check availability

- If $\text{ngo} > \text{navail}$ then $\text{ngo} = \text{navail}$
- Else $\text{ngo} = \text{ngo}$

**Step3:** Update number of molecules

- $r_i0 = r_i0 - (\text{ngo} \times r_s0)$
- $r_i1 = r_i1 - (\text{ngo} \times r_s1)$
- $p_i0 = p_i0 + (\text{ngo} \times p_s0)$
- $p_i1 = p_i1 + (\text{ngo} \times p_s1)$
**Output Files**

All output files are written according to this flowchart. First the time is written, then the names of all species along with the name of its sub-compartment are written, and then time value and concentration or number of molecules is put. It is difficult to analyze the NeuroRD output in text format. We can also view the graphical output using a tool NeuroRD_Output_Analyzer. We can visualize the specific species concentrations/number of molecules according to sub-compartment, dendrite or whole neuron.

*Figure 29: Output structure of NeuroRD*

NeuroRD also produce a mesh output file where the geometrical position (coordinates of four points of a compartment) of each compartment described with their volume after discretization.

*Figure 30: Sample Graphical Output of NeuroRD using the tool NeuroRD_Output_Analyzer*

---

2 NeuroRD_Output_Analyzer is a tool developed in Matlab by me for graphical output of NeuroRD.
4.2.2 Stochastic Simulation through NeuroRD

Membrane bound species and free species both are widely used in biochemical signaling networks. In this phase, I have investigated a simple cascade with membrane bound and free species using the stochastic simulator, NeuroRD. I have considered a toy model which is included in Appendix D. In this toy model, I considered a piece of the dendrite, together with a simple cascade, $A + B \leftrightarrow C + D$. The aim is to observe the mobility of membrane bound species and free species in neurons through NeuroRD, and also to get a feeling for pitfalls.

**Experiment 1:**

The first case is organized as follows with diffusion constant 0 for every species, fixed time step 0.0005, total number of compartments 100, forward and backward rate constant 1.

<table>
<thead>
<tr>
<th>Species</th>
<th>Initial Concentration/Density</th>
<th>Before Simulation</th>
<th>After Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#molecules</td>
<td>#molecules</td>
<td>#molecules</td>
</tr>
<tr>
<td>A</td>
<td>45</td>
<td>54</td>
<td>28</td>
</tr>
<tr>
<td>B</td>
<td>45</td>
<td>54</td>
<td>28</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>26</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>26</td>
</tr>
</tbody>
</table>

*Figure 31: Experiment 1, Species in dendrite (left) and the quantity table (right)*

This was tested to see the system seems to be in chemically equilibrium state.

**Experiment 2:**

The second case is organized as follows with the same parameter as in the first experiment. The difference from the first experiment is that no membrane bound-species exist here.

<table>
<thead>
<tr>
<th>Species</th>
<th>Initial Concentration/Density</th>
<th>Before Simulation</th>
<th>After Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#molecules</td>
<td>#molecules</td>
<td>#molecules</td>
</tr>
<tr>
<td>A</td>
<td>140</td>
<td>53</td>
<td>44</td>
</tr>
<tr>
<td>B</td>
<td>140</td>
<td>57</td>
<td>48</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
</tbody>
</table>

*Figure 32: Experiment 2 environment (left) and the result table (right)*

Now, after simulation the system does not show the state of chemical equilibrium. Is it only for absence of membrane-bound species or not? Seems the system $A + B$ cannot produce sufficient number of $C + D$. 

41
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

**Experiment 3:**

To get the answer of question from experiment 2, this experiment is organized as experiment 2 except total number of compartments. Here it is 36 instead of 100.

<table>
<thead>
<tr>
<th>Sub-membrane</th>
<th>Species</th>
<th>Initial Concentration/Density</th>
<th>Before Simulation</th>
<th>After Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cytosol</td>
<td>A, B, C, D</td>
<td>140</td>
<td>48</td>
<td>28</td>
</tr>
<tr>
<td>Sub-membrane</td>
<td>C</td>
<td>0</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>0</td>
<td>0</td>
<td>20</td>
</tr>
</tbody>
</table>

*Figure 33: Experiment 3 environment (left) and the result table (right)*

Experiment 3 seems to be in the state of chemical equilibrium. According to this experiment, chemical equilibrium is depending on number of compartments. In fewer numbers of compartments, the number of idle (one molecule alone in one compartment) molecule is less than before.

**Experiment 4:**

This experiment is also arranged as experiment 2 but in here I used diffusion constant 10.0 instead of 0 for species C and D. To maintain the leap condition, I also decreased the fixed time step from 0.0005 to 0.00005. The difference between experiment 3 and 4 is that, in experiment 3 I decreased the number of compartments but in experiment 4, I introduced diffusion constant instead of reducing number of compartments.

<table>
<thead>
<tr>
<th>Sub-membrane</th>
<th>Species</th>
<th>Initial Concentration/Density</th>
<th>Before Simulation</th>
<th>After Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cytosol</td>
<td>A, B, C, D</td>
<td>140</td>
<td>53</td>
<td>27</td>
</tr>
<tr>
<td>Sub-membrane</td>
<td>C</td>
<td>0</td>
<td>57</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>0</td>
<td>0</td>
<td>26</td>
</tr>
</tbody>
</table>

*Figure 34: Experiment 4 environment (left) and the result table (right)*

In this case, system also seems to be in the state of chemical equilibrium.

That means system equilibrium is not only depending on the number of compartments but also on diffusion constant. Due to higher diffusion rate there is a possibility to decrease the number of idle (one molecule alone in one compartment) molecules. Another finding is that to maintain probability of event of one particle, p in NeuroRD, one may need to change the fixed time steps. Above experiments have been done for small number of molecules. Now we can go through the same tests for large number of molecules.
Experiment 5:

This experiment is same as experiment 1 but with large number of molecules and rate constant 0.01.

<table>
<thead>
<tr>
<th>Species</th>
<th>Initial Concentration/Density</th>
<th>Before Simulation</th>
<th>After Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>900</td>
<td>1083</td>
<td>527</td>
</tr>
<tr>
<td>B</td>
<td>900</td>
<td>1083</td>
<td>527</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>556</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>556</td>
</tr>
</tbody>
</table>

Figure 35: Experiment 5 environment (left) and the result table (right)

Here the system seems to be in chemically equilibrium state.

Experiment 6:

This experiment is same as experiment 2 but for large number of molecules and rate constants 0.01.

<table>
<thead>
<tr>
<th>Species</th>
<th>Initial Concentration/Density</th>
<th>Before Simulation</th>
<th>After Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>14000</td>
<td>5062</td>
<td>2532</td>
</tr>
<tr>
<td>B</td>
<td>14000</td>
<td>5068</td>
<td>2538</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>2530</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>2530</td>
</tr>
</tbody>
</table>

Figure 36: Experiment 6 environment (left) and the result table (right)

This also seems to be in chemically equilibrium state for both membrane-bound species and free species with large number of population. To be more confident about the systems chemical equilibrium and it’s dependencies I investigate more on same model.

Then, I have done the same experiments with same model but using simpler cascade $A + B \leftrightarrow C$. It’s also gives the similar result. But in this case, to test the chemically equilibrium state of the system from number of molecules one has to do the calculations as follows:

\[
\text{molecules for species } A \times \frac{\text{molecules for species } B}{0.6022} = \text{molecules for } C
\]
4.3 Discussion

The objective of this thesis had three aspects: first, to observe the nature of a feed-forward loop in a deterministic model, second, to retrieve the structure and behavior of NeuroRD and third, to investigate the behavior of membrane-bound species and free species as well as the number of molecules in neuronal signaling cascades using NeuroRD.

To better understand the deterministic modeling of signaling cascades, I observed the feed-forward loop PKAc-PP2A-pThr75 in the original Lindskog model and compared it with a more recently published model by Nakano. This feed-forward loop was described in the Lindskog model as a negative feed-forward loop and in the Nakano model as a positive feed-forward loop. Both models are quite similar. However, according to the Nakano model PP2A cannot dephosphorylate DARPP32 but in the Lindskog model PP2A can dephosphorylate DARPP32. I investigated this feed-forward loop through the Lindskog model and the Nakano model (by adjusting the Lindskog model).

During the investigation I used transient or steady-state input and with different amplitudes. With transient and steady-state input for both low (1000) and high (20000) amplitude the key findings shown below:

- PKAc is always small when the loop is intact. This means PKAc-PP2A-pThr75 feed-forward loop behaves as a negative feed-forward loop.

- pThr34 either follow PKAc, or if dopamine input is large or has a long duration. pThr34 can be higher if the loop is intact. This is because the balance between pThr34 and pThr75 is shifted.

- During dopamine transient inputs, only pThr34 depends on amplitude with the original Lindskog model. Here, the dopamine amplitude is proportional to pThr34 when loop is intact. But with the adjusted Lindskog model pThr34 always small when loop is intact.

- During dopamine steady-state input, only DARPP32 depends on amplitude with the original Lindskog model. Here, the dopamine amplitude is reversely proportional to DARPP32 when loop is intact. But with the adjusted Lindskog model DARPP32 always large when loop is intact.

Surprisingly, these investigations found that PKAc-PP2A-pThr75 feed-forward loop behaves as a negative feed-forward loop in both models if investigated in similar way.

Then NeuroRD, a tool for simulating models such as Lindskog or Nakano, was tested. Input files, output files, the structure and flow of NeuroRD are explained in section 4.2.1 with various graphs and figures. The list of findings and source code references are included in Appendix A and B respectively. The prerequisites on how to simulate biochemical pathway and graphical outputs of NeuroRD are also discussed there. That’s the second aspects of this thesis. Several ‘bugs’ were discovered during the investigation and testing phase. Feedbacks are provided to the developers.
In the third and final part of this thesis, a simple cascade like $A + B \leftrightarrow C + D$ was investigate in detail using NeuroRD. Several insights were gained. According to the NeuroRD equilibrium of the system depends on:

i. Fixed time steps  
ii. Number of compartments  
iii. Diffusion rate  
iv. Rate constants

Even, by using simpler cascade $A + B \leftrightarrow C$, the equilibrium of the system also can be defined through these above four parameters.
Chapter 5: Future Work

After completing my thesis work, I felt that more investigations are needed to solve some key issues in the near future. In the deterministic part, I have shown that the feed-forward loop PKA-PP2A-Thr75 behaves as a negative feed-forward loop. I investigated this loop according to the Lindskog model and the Nakano model (by adjusting the Lindskog model). In the Nakano paper this PKAc-PP2A-pThr75 loop behaves as a positive feed-forward loop. The question is why this loop is showing different behavior? This would require an implementation of their model.

In NeuroRD, the explicit tau-leap has been implemented but implementation of implicit tau-leap can be the next future task to get faster and efficient stochastic simulator.

More investigation should be performed to define the chemical equilibrium of a system through NeuroRD using free species and membrane-bound species.

NeuroRD could be connecting with electrical simulator like MOOSE to run both electrical and chemical signaling in parallel. This work is in progress [Accepted Abstract in CNS*2011 included in Appendix C].

Parallel computing is growing and developing rapidly, therefore it could be the next task to design and development a biochemical simulator for parallel computing.
Bibliography


47
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.


Simulating receptor induced cascades in neurons – with special reference to basal ganglia.


Appendix A

List of –findings related to NeuroRD:

(“Bugs” have been regarded to the developer)

1. “defaultMaxElementSide” property of Master or model file works properly but the property “MaxElementSide” cannot work because there is no verification for this property. Sometimes it works when the user can set the exact input value.

2. The property “outputQuantity” of model file has no effect in NeuroRD, though it should produce the result in the number of molecules instead of concentrations.

3. The rules of labeling dendrite or spine for identifying the injection site are not precisely described. These rules are described in this paper at section 4.2.1 (Morphology section).

4. “lengthDensity” and “areaDensity” are properties of Morphology input files. These properties should calculate the average number of spines according to per micron of dendrite length and per unit area. But NeuroRD always calculate the average number of spines according to per unit area, in accordance with the set property; “lengthDensity” or “areaDensity”.

5. According to NeuroRD, in a 2D dendrite and after compartmentalization, one compartment can have maximum four neighbors because diagonal neighbor are not considered here. See section 4.2.5

6. A dendritic stick has two regions; one is the sub-membrane region and the second one is cytosolic region. After discretization of the dendrite, these two regions should exist. But if you subdivide the dendrite in smallest numbers of compartments, then NeuroRD can hold only one sub-membrane region. Specifically, when max element side is equal to the width of dendrite and we want to produce a few numbers of compartments, then the compartments hold only one sub-membrane region instead of two. This means that molecules are of 50% of the expected our.

7. To avoid the warning “p too large…..” we have to either reduce the time step (“fixedStepDt” property in model file) or reduce the number of compartments. This is the test of the tau-leap condition of Gillespie’s algorithm. NeuroRD warns, but then just continues.

8. Data verification, error and warning messages will not be easily perceived by the ordinary users, unless they have a significant amount of knowledge in this field.
Appendix B

Figure 37: Working diagram of NeuroRD with source code reference.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

*1. All information reads and sets through the following classes:

- SDRun.java
- Morphology.java
- ReactionScheme.java
- OutputScheme.java
- StimulationSet.java
- InitialConditions.java

[All these classes extracted from, package org.textensor.stochdiff.model;]

*2. For discretization of dendrite the following classes and functions also used:

- BaseCalc.java [package org.textensor.stochdiff.numeric;]
  extractGrid()
- TreeBoxDiscretizer.java [package org.textensor.stochdiff.disc;]
  [Only for discretization]
  TreeBoxDiscretizer()
  buildGrid()
- Discretization.java [package org.textensor.stochdiff.model;]
  [When the properties “MaxElementSide” used]
  getResolutionHM()
- SegmentSlicer.java [package org.textensor.stochdiff.disc;]
  getFixedWidthSlices()
  discretized()
  getslices()
- LineBoxer.java [package org.textensor.stochdiff.disc;]
  [For 2D dendrite]
- DiscBoxer.java [package org.textensor.stochdiff.disc;]
  [For 3D dendrite]

For Spine discretization:

- SpineLocator.java [package org.textensor.stochdiff.disc;]

For concentration and density of species for each compartment:

- BaseCalc.java [package org.textensor.stochdiff.numeric;]
  makeRegionConcentrations()
  makeRegionSurfaceDensities()
- VolumeGrid.java [package org.textensor.stochdiff.numeric.morph;]
  getRegionLabels()

*3. SDCalc.java [package org.textensor.stochdiff;] class called SteppedStochasticGridCalc.java class to run the simulation through BaseCalc.java.

Calculate number of molecules: wkA[compartment#][species#] array store the number of molecules which calculated in init() method of SteppedStochasticGridCalc.java class.
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

*4. To inject any molecule during simulation can use advance(time) method of SteppedStochasticGridCalc.java class.

*5. To write the output the following functions of SteppedStochasticGridCalc.java class used:

- getGridConcsHeadings_dumb()
- getGridConcsPlainText_dumb()

N.B This is not the complete reference of the source code, but it will guide you to understand the source code of NeuroRD.
Appendix C

Connecting MOOSE and NeuroRD through MUSIC: Towards a communication framework for multi-scale modeling

Maya Brandi¹,², Ekaterina Brocke²,³,⁴, Husain Ahammad Talukdar³, Michael Hanke², Upinder S. Bhalla¹, Jeanette Hellgren Kotaleski²,³,⁴, Mikael Djurfeldt¹,⁴

¹ PDC, Royal Institute of Technology - KTH, Stockholm, S-100 44, Sweden
² CSC, Royal Institute of Technology - KTH, Stockholm, S-100 44, Sweden
³ National Centre for Biological Sciences, Bangalore, India
⁴ INCF, Karolinska Institutet - KI, Stockholm, S-171 77, Sweden
E-mail: djurfeldt@incf.org

The nervous system encompasses structure and phenomena at different spatial and temporal scales from molecule to behavior. In addition, different scales are described by different physical and mathematical formalisms. The dynamics of second messenger pathways can be formulated as stochastic reaction-diffusion systems [1] while the electrical dynamics of the neuronal membrane is often described by compartment models and the Hodgkin-Huxley formalism. In neuroscience, there is an increasing need and interest to study multi-scale phenomena where multiple scales and physical formalisms are covered by a single model. While there exists simulators/frameworks, such as GENESIS and MOOSE [3], which spans such scales (kinetikit/HH-models), most software applications are specialized for a given domain. Here, we report about initial steps towards a framework for multi-scale modeling which builds on the concept of multi-simulations [2]. We aim to provide a standard API and communication framework allowing parallel simulators targeted at different scales and/or different physics to communicate on-line in a cluster environment. Specifically, we show prototype work on simulating electrical activity and Ca²⁺-dynamics in a dendritic spine using MOOSE and NeuroRD [4,8].

Electrical properties of a simple compartment model with soma, dendrite and spine is simulated in MOOSE, while Ca²⁺ dependent reactions and diffusion in the spine is simulated in NeuroRD. In a prototype system, the two simulators are connected using PyMOOSE [5] and JPype [7]. The two models with their different physical properties (membrane currents in MOOSE, biophysics in NeuroRD), are joined into a single model system. Ca²⁺ currents in the electrical model are translated to Ca²⁺ influx rates in NeuroRD, determining the dynamics of the biophysical model. In turn Ca²⁺ dependent events in the spine control properties such as Ca²⁺ dependent ion channels in the electrical model. The joint system, including details of solver methods, is also studied analytically with regard to stability and accuracy and a set of requirements for a generic API allowing parallel solvers to communicate in a multi-simulation is formulated. Experiences from couplers [6] used to couple field models in climate research is taken into consideration. A gap analysis with respect to the existing MUSIC framework [2] is performed.

We demonstrate the interaction of the numerical solvers of two simulators (MOOSE, NeuroRD) targeted at different spatiotemporal scales and different physics while solving a multi-scale problem. We analyze some of the problems that may arise in multi-scale multi-simulations and present requirements for a generic API for parallel solvers. This work represents an initial step towards a
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

flexible modular framework for simulation of large-scale multi-scale multi-physics problems in neuroscience.

References:


7. Jpype Bridging the worlds of Java and Python [http://jpype.sourceforge.net/]

8. NeuroRD [http://krasnow.gmu.edu/CENlab/software.html]
Appendix D

Six input files of the “toymodel” are used given below:

1. Model file

```xml
<SDRun>

<!-- this file defines a single run of the calculation, using morphology and
reaction data brought in from other files -->

<reactionSchemeFile> MSPNreactions_new_1spine </reactionSchemeFile>
<morphologyFile> MSPNmorph_1spinea </morphologyFile>
<stimulationFile> MSPNstim_5DaStim </stimulationFile>
<initialConditionsFile> MSPNic_A </initialConditionsFile>
<outputSchemeFile> MSPNio_1spine_newmorph </outputSchemeFile>

<!-- 2D means the morphology is interpreted like a flatworm, 3D for
roundworms. The 2D case is good for testing as it is easy to visualize the
results (also, 3D may not work yet...) -->

<geometry> 2D </geometry>
<depth2D> 0.4 </depth2D>
<distribution> BINOMIAL </distribution>
<algorithm> INDEPENDENT </algorithm>
<simulationSeed> 0000123 </simulationSeed>

<!-- run time for the calculation, milliseconds -->
<runtime> 2500 </runtime>

<!-- set the seed to get the same spines each time testing -->
<spineSeed> 123 </spineSeed>
```

56
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

<discretization>

<!-- default largest size for elements in bulk volumes (dendrites), microns -->
<defaultMaxElementSide>0.12</defaultMaxElementSide>
<!--defaultMaxElementSide>0.2</defaultMaxElementSide-->
<!-- discretization for spines, microns -->
<brineDeltaX> 0.1 </brineDeltaX>

</discretization>

<!-- timestep used in fixed step calculations, in milliseconds -->
<fixedStepDt> 0.0005 </fixedStepDt>
<!-- interval at which stuff should be saved to the output file(s) -->
<outputInterval> 3000000.0 </outputInterval>
<!-- the tolerace is not used yet -->
<tolerance> 0.001 </tolerance>
<!-- calculation types include GRID_STEPPED_STOCHASTIC and GRID_STEPPED_CONTINUOUS for reaction-diffusion systems. Single mixed pool calculations should be listed here too (TODO) -->
<calculation>GRID_STEPPED_STOCHASTIC</calculation>

</SDRun>
2. Initial Condition file:

```xml
<InitialConditions>
  <!-- these apply to everything unless overridden -->
  <ConcentrationSet>
    <!-- For small number molecules -->
    <!-- NanoMolarity specieID="A" value="140" /> -->
    <NanoMolarity specieID="B" value="140"/>
    <!-- For large number molecules -->
    <NanoMolarity specieID="A" value="14000"/>
    <NanoMolarity specieID="B" value="14000"/>
    <NanoMolarity specieID="C" value="0"/>
    <NanoMolarity specieID="D" value="0"/>
  </ConcentrationSet>
  <!-- This is the block of SurfaceDensitySet applied to the dend region -->
  <!-- SurfaceDensitySet region="dend1"> -->
    <PicoSD specieID="A" value="45.00"/>
    <PicoSD specieID="B" value="45.00"/>
    <PicoSD specieID="A" value="900.00"/>
    <PicoSD specieID="B" value="900.00"/>
    <PicoSD specieID="D" value="0"/>
  </SurfaceDensitySet>
</InitialConditions>
```
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

3. Output file

```xml
<OutputScheme>
  <OutputSet filename = "cell0" dt="100.0">
    <OutputSpecie name="A" />
    <OutputSpecie name="B" />
    <OutputSpecie name="C" />
    <OutputSpecie name="D" />
  </OutputSet>
</OutputScheme>
```

4. Morphology file

```xml
<Morphology>
  <Segment id="seg1" region="dend1">
    <start x="1.0" y="1.0" z="0.0" r="0.3"/>
    <end x="3.5" y="1.0" z="0.0" r="0.3"/>
  </Segment>
</Morphology>
```
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.

5. Reaction file

```xml
<ReactionScheme>
  <Specie name="A" id="A" kdiff="0" kdiffunit = "mu2/s"/>
  <Specie name="B" id="B" kdiff="0" kdiffunit = "mu2/s"/>
  <Specie name="C" id="C" kdiff="0" kdiffunit = "mu2/s"/>
  <Specie name="D" id="D" kdiff="0" kdiffunit = "mu2/s"/>
  <!-- PULSE: A + B <-> C + D-->
  <Reaction name = "A+B--C+D reac" id="A+B--C+D_id">
    <Reactant specieID="A" />
    <Reactant specieID="B" />
    <Product specieID="C" />
    <Product specieID="D" />
    <forwardRate> 1 </forwardRate>
    <reverseRate> 1 </reverseRate>
    <Q10> 0.2 </Q10>
  </Reaction>
</ReactionScheme>
```

6. Stimulation file

```xml
<StimulationSet>
  <!-- Inject nothing-->
</StimulationSet>
```
Simulating receptor induced cascades in neurons – with special reference to basal ganglia.