DEGREE PROJECT, IN COMPUTER SCIENCE, DEGREE PROGRAM IN
ENGINEERING PHYSICS (CTFYS) 270 CR , SECOND LEVEL
STOCKHOLM, SWEDEN 2014

Normalization in a cortical hypercolumn

THE MODULATORY EFFECTS OF A HIGHLY STRUCTURED RECURRENT SPIKING NEURAL NETWORK

YLVA JANSSON

KTH ROYAL INSTITUTE OF TECHNOLOGY

COMPUTER SCIENCE AND COMMUNICATIONS (CSC)

Normalization in a cortical hypercolumn

- the modulatory effects of a highly structured recurrent spiking neural network

Normalisering i en kortikal hypercolumn

- modulerande effekter i ett hårt strukturerat rekurrent spikande neuronnätverk

Ylva Jansson

DD221X, Degree project in Computer Science, second level, 30 cr
Degree Program in Engineering Physics (CTFYS), 270 cr
Royal Institute of Technology, year 2014
Supervisor: Prof. Anders Lansner
Examiner: Prof. Jens Lagergren

Royal Institute of Technology
School of Computer Science and Communication

KTH CSC
SE-100 44 Stockholm, Sweden
URL:www.kth.se/csc

To Iris and Spiskummin for relentlessly sleeping on my computer and staring at me from the bookshelf, until all problems eventually and inevitably are solved.

Abstract

Normalization is important for a large range of phenomena in biological neural systems such as light adaptation in the retina, context dependent decision making and probabilistic inference. In a normalizing circuit the activity of one neuron/-group of neurons is divisively rescaled in relation to the activity of other neurons/groups. This creates neural responses invariant to certain stimulus dimensions and dynamically adapts the range over which a neural system can respond discriminatively on stimuli. This thesis examines whether a biologically realistic normalizing circuit can be implemented by a spiking neural network model based on the columnar structure found in cortex. This was done by constructing and evaluating a highly structured spiking neural network model, modelling layer 2/3 of a cortical hypercolumn using a group of neurons as the basic computational unit. The results show that the structure of this hypercolumn module does not per se create a normalizing network. For most model versions the modulatory effect is better described as subtractive inhibition. However three mechanisms that shift the modulatory effect towards normalization were found: An increase in membrane variance for increased modulatory inputs; variability in neuron excitability and connections; and short-term depression on the driving synapses. Moreover it is shown that by combining those mechanisms it is possible to create a spiking neural network that implements approximate normalization over at least ten times increase in input magnitude. These results point towards possible normalizing mechanisms in a cortical hypercolumn; however more studies are needed to assess whether any of those could in fact be a viable explanation for normalization in the biological nervous system.

Sammanfattning

Normalisering är viktigt för en lång rad fenomen i biologiska nervsystem såsom näthinnans ljusanpassning, kontextberoende beslutsfattande och probabilistisk inferens. I en normaliserande krets skalas aktiviteten hos en nervcell/grupp av nervceller om i relation till aktiviteten hos andra nervceller/grupper. Detta ger neurala svar som är invarianta i förhållande till vissa dimensioner hos stimuli, och anpassar dynamiskt för vilka inputmagnituder ett system kan särskilja mellan stimuli. Den här uppsatsen undersöker huruvida en biologiskt realistisk normaliserande krets kan implementeras av ett spikande neuronnätverk konstruerat med utgångspunkt från kolumnstrukturen i kortex. Detta gjordes genom att konstruera och utvärdera ett hårt strukturerat rekurrent spikande neuronnätverk, som modellerar lager 2/3 av en kortikal hyperkolumn med en grupp av neuroner som grundläggande beräkningsenhet. Resultaten visar att strukturen i hyperkolumnmodulen inte i sig skapar ett normaliserande nätverk. För de flesta nätverksversioner implementerar nätverket en modulerande effekt som bättre beskrivs som subtraktiv inhibition. Dock hittades tre mekanismer som skapar ett mer normaliserande nätverk: Ökad membranvarians för större modulerande inputs; variabilitet i excitabilitet och inkommande kopplingar; och korttidsdepression på drivande synapser. Det visas också att genom att kombinera dessa mekanismer är det möjligt att skapa ett spikande neuronnät som approximerar normalisering över ett en åtminstone tio gångers ökning av storleken på input. Detta pekar på möjliga normaliserande mekanismer i en kortikal hyperkolumn, men ytterligare studier är nödvändiga för att avgöra om en eller flera av dessa kan vara en förklaring till hur normalisering är implementerat i biologiska nervsystem.

Preface

The subject of this thesis lies in the intersection of mathematics, information theory, biology, psychology and artificial intelligence. The project have given me the opportunity to peek into some of the still unanswered questions in these intriguing fields and get acquainted with cutting edge theories of neuroscience and computational neuroscience. This together with getting the chance to contribute to solving one of the many unanswered questions, have without competition been the most exciting, interesting and rewarding experience throughout my studies. I wish to thank my supervisor Anders Lansner for allowing me to follow my intuition and asking the right questions. The spirit of the computational biology group at KTH which can be summarized as “we don’t necessarily do as everyone else here” have been truly inspiring. Thanks to Pavel, Phil, Florian, Pierre, Bernard, Matthias, Pradeep, Örjan, Arvind, Tony and Georgios for sharing their ideas, and questioning as well as encouraging mine. I have really felt accepted into the research group as one of their own. The undertaking of this project have been to be lost and found and then lost again, as new layers of complexity unfold. The hard part have not been to find interesting things to investigate but to not get carried away to too many distant and alluring places at once. My hope is that this thesis provide some answers and asks some questions that will give a small contribution to understanding how normalization might be implemented in a cortical hypercolumn.

Contents

List of abbreviations and terminology

ANN	Artificial neural network
EPSP	Excitatory postsynaptic potential
HC	Hypercolumn
IPSP	Inhibitory postsynaptic potential
FIR test	Fixed Input Relations test. Test designed in this thesis for evaluating normalization by considering how an input vector of fixed input relations is processed depending on input magnitude.
IAF neuron	Integrate-and-fire neuron
IO test	Input - output test. Test for evaluating normalization where the effect of a modulatory input on a single computational units IO-curve is studied.
IO curve	Output firing rate for computational unit (here neuron/minicolumn) as function of input rate. Also termed “Input - output curve” and “response curve”.
NEST	The Neural Simulation Tool. Simulator for spiking neural network models used in this thesis.
PSP	Postsynaptic potential
STD	Short-term depression
V1	Primary visual cortex
WTA	Winner-take-all
MC/mc	Minicolumn. Group of recurrently connected cells found in primate cortex and used as the basic computational unit for models implemented in this thesis.
Modulatory effect	How the response function of a single unit is affected by a modulatory input.
Modulatory input	A secondary input that will affect how a neuron/-computational unit processes a driving or primary input. E. g. for a normalizing circuit the modulatory input would be the input to the other computational units in the circuit.

1 Introduction

Understanding the mysteries of the brain is often described as “one of the last scientific frontiers” together with the origin of our universe. When the research on the origin of it all is about understanding how we came to be here, neuroscience touches the questions about who we really are and why we are the way we are. The human brain constantly performs tasks that might *seem* mundane since we seemingly effortless perform them every day but are in fact highly computationally complex. How are we able to recognize our tea cup or coworker, decipher sounds to words and understand the meaning of those words? How can we decide what to have for dinner, coordinate limb movements to cook said dinner or integrate information from different senses to shape a coherent representation of the kitchen?

Even though the great efforts put into solving these questions unquestionably have resulted in significant advances in neuroscience and computational neuroscience over the last decades, there are still large parts of the puzzle missing to satisfyingly answer how the brain performs these tasks. An improved understanding of the biological nervous system could (aside from it being extremely interesting in its own right), improve diagnosis and treatment for psychiatric disorders and brain diseases (Lansner 2009) and give us a deeper understanding of human and animal behavior. Moreover it could give way to a new generation of neuromorphic computers and artificial intelligence systems using the same principles for computation as the biological nervous system.

Could finding the missing pieces and putting them together, also bridge the gap between how the brain and billions of neurons works on a low level scale, and how we experience consciousness, sense of self and being distinct from the world around us? An often cited quote attributed to various sources states:

> *“If the brain were so simple we could understand it, we would be so stupid that we couldn’t”*

This harbors an intriguing question: Will it be possible to use our brain as an instrument to dissect itself? Will we succeed in our endeavor to unravel the mysteries of this complex biological system that defines who and what we are? (Cannon 2014). Maybe in a few decades from now we will actually know the answer to that question.

Modeling the Brain

This brings us to the question of how to find the missing pieces of the puzzle. One way to get a better understanding of the brain is to make a model of it. In a model it is straight forward to perform experiments not feasible or ethical to perform on a biological brain. Parameters can be changed to understand what significance they have, the scale of the model can be reduced or increased, the connectivity, neuron type and synapses can be modified as we wish. Biologically realistic models of the nervous system can thus help bridge the gap between our understanding of behavior and psychology in a top down way, and our bottom up understanding of the morphology and functionality of neural circuits and their information processing abilities. It can also help reducing the amount of costly and morally dubious animal studies needed for solving these tasks. Developing computer hardware or software based on the way the brain processes information (stochastic, local, distributed), that transcends the synchronized deterministic way of a standard computer of today, is also thought to give way to more powerful and energy efficient computers that could solve problems which are currently intractable. Computer science and computational neuroscience thus gets interleaved: The former enable us to construct the complex and large scale computer models necessary to understand the biological nervous system; and a better understanding of the biological nervous system could give way to a new generation of computers and artificial intelligence applications.

The third generation of neural networks with biologically realistic spiking neurons are because of the high degree of similarity to real neurons, especially suited for these tasks (Ponulak & Kasinski 2011). With spiking neural networks it is indeed possible to gain deeper insights into how a biologically realistic neural network can perform e. g. sensory processing, memory formation and decision making.

Canonical neural computations

From a computational perspective the brain can be understood as organized in a large number functional units or modules, where data is processed and transformed creating successively more complex representations of the surrounding world. It has been shown that a set of distinct neural computations appear as parts in information processing for disparate modalities, tasks, and species (Carandini & Heeger 2012). Understanding these “canonical neural computations” is an important part in deciphering the brains information processing. Exponentiation (non-linear transformation of incoming our outgoing signals and linear filtering (weighed summation by linear receptive fields) have long been recognized as canonical neural computations. Normalization where the activity in one computational unit is rescaled in relation to the activity in other computational units, is now proposed to be another one because of its presence in multiple neural systems (Carandini & Heeger 2012).

Normalization

Normalization is a nonlinear cortical operation where the activity of a single neuron/computational unit is rescaled in relation to the total activity in a group of neurons/units. This will lead to a disregard of the absolute input magnitude to the system, while preserving information about the *relation* between different inputs. In several experimental studies, see for example the primate visual system (Carandini, Heeger 1997) and the olfactory system of the fruit fly (Olsen et al. 2010), the below stated relation between direct input to a single neuron and input to the normalization pool (the group of neurons which rescale the output) is found.

$$R_j = R_{max}\frac{D_j^n}{\sigma^n + \sum D_k^n} \tag{1.1}$$

Here R_j is activity in neuron j; D_j driving input to neuron j; D_k activity for neuron k in the normalization pool; and R_{max}, σ and n experimentally derived constants. The important thing to note, is how increased activity in the normalization pool suppress the output from a single neuron *in a divisive way*. For processing of an input vector of the form $\overline{x} = [ac_1, ac_2, .. ac_n]$ normalization would mean that irrespectively of the magnitude of the input vector - that is for different choices of a - the summed output from the system as well as the relations between the outputs will be kept approximately constant. Normalization can thus be used to create a neuronal circuit invariant or partially invariant to certain stimulus dimensions.

These features make normalization significant for effective information processing, for example by dynamically adapting the range in which a cell can respond discriminatively on input which is illustrated by a toy example in Figure 1.1, and facilitate discrimination by a linear classifier (Olsen et al. 2010). It is also suggested as an explanation for how attention affects the interpretation of sensory inputs (Reynolds & Heeger 2009), and proposed as an important mechanism in multisensory integration (van Atteveldt et al. 2014) and decision making (Louie et al. 2013). Normalization is also hypothesized to play a role for the type of probabilistic inference in neural networks known as marginalization (Beck et al. 2011).

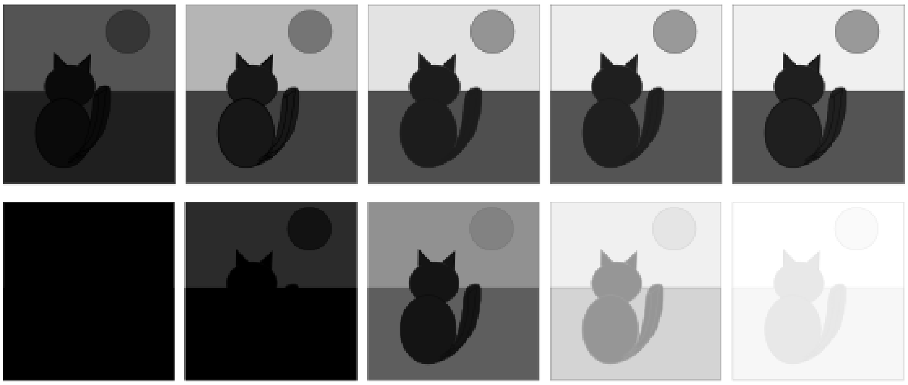

Figure 1.1: Toy example of how normalization in the visual system preserves the relative light intensities in a scene. A greyscale image for a scene with increasing average light intensity is visualized as the output from (top row) The normalization model (Equation 1.1) and (bottom row) a neuronal response function without the divisive scaling in relation to the other inputs (Equation 1.1 with the sum in the denominator replaced with D_j^n). Average light intensity is increased from right to left using scale factors 1/10.0, 1/√10.0, 1.0, √10.0 and 10.0. Input vector for scale factor 1.0: I = [100, 300, 500, 1000].

The phenomenological studies that show the presence of normalization in biological nervous systems are abundant and result in strikingly similar mathematical models that describe neuronal responses. However the biological mechanisms underlying normalization on cell or network level remain elusive.

Proposed mechanistic explanations for normalization

Among proposed mechanisms for implementing normalization is varying amounts of balanced (no net excitatory or inhibitory drive) noise which is shown to result in a gain change for the response curve of a single in vitro neuron (Chance et al. 2002), modulation by ongoing activity of the brain that affect subthreshold membrane potentials (Carandini & Heeger 2012), short-term synaptic depression and/or non-linear effects in the dendrites or afferent inputs (Silver 2010). Experimental or theoretical studies have shown that these mechanisms can give rise to divisive gain control under certain conditions, but it has not yet been proved which one of them, if any, that are active in shaping normalization for real cortical circuits and neurological tasks.

As to modeling in silico, gain control through balanced background activity (Ayaz & Chance 2009, Ly & Doiron 2009), short-term synaptic depression (Carandini et al. 2002, Rothman et al. 2009) and the effect of nonlinearities in the afferent inputs (Murphy & Miller 2003) have been studied in more or less complex models of single neurons. Winner-take-all dynamics have also been implemented in a biologically realistic attractor memory model (Lundqvist et al. 2006). However it is to our knowledge not yet studied how normalization can be implemented *intrinsically* in a larger biologically realistic spiking neural network using a group of neurons as the computational unit.

A better understanding of the mechanisms that give normalization in spiking neural networks would be important both for creating larger biologically realistic networks composed of different modules, and for designing effective information processing networks e. g. for artificial intelligence tasks. It would also provide guidance as to the mechanisms for normalization in biological nervous systems which could guide further research into this important but elusive phenomenon.

1.1 Aim of study

The purpose of this thesis is to gain insight into how normalization can be implemented in a biologically realistic way by trying to design a spiking neural network that implements normalization. The starting point is several proposed mechanistic explanations for normalization in biological nervous systems and an existing spiking neural network model of layer 2/3 of a cortical hypercolumn – a highly structured network with recurrent connectivity and a group of interconnected neurons as the basic computational unit.

The project is constrained to achieve normalization within the means of realistic neuron models and biological mechanisms represented in these. The overarching hoped for result is twofold: To gain increased understanding of normalization in spiking neural nets and more specifically this type of highly structured recurrent network; and to gain insight into how normalization might be implemented in biological nervous systems.

More specifically this thesis aims to answer the following questions:

1. What kind of modulatory effects can be implemented by the hypercolumn model? The functionality is compared to three well known models of cortical processing:
    a) Subtractive inhibition
    b) Output gain
    c) Normalization
2. What are the roles of the feed-back and feed-forward inhibition in the network?
3. How does changing the parameters of the system affect the kind of modulation it performs, and is it possible to implement normalization by varying the parameters within a biologically plausible parameter range?
4. If so over what range of input magnitudes is the system normalizing?
5. What limits the system’s ability for normalization?
6. Will adding synaptic depression to the system, change how it modulates its input and if so in what way?
7. Could the structure of the hypercolumn model, be a plausible mechanistic explanation of normalization in biological nervous systems?

2 Preliminaries

2.1 Outline of report

This chapter “Preliminaries” revisits preliminaries of biological neurons and nervous systems, artificial neural networks as well as spiking neural networks and neuron models. It can be skipped by a reader already familiar with these subjects. The third chapter “Theory” summarizes the current state of art considering information coding in the brain and spiking neural networks, as well as normalization in spiking neural networks and biological nervous systems. This is the scientific basis for the investigation conducted in this thesis and the model construction. The fourth chapter “Materials and methods” describes the blueprint for the network model – the hypercolumn module - and gives all details of the implemented model versions. The second part of this chapter “Evaluation methods” presents the three theoretical models of cortical modulatory modes used as a reference during evaluation. It also details how the normalization abilities of the model versions were assessed. In the fifth chapter “Results” the reader will be guided through the evaluation of the hypercolumn module and the performance of different model versions are presented. The last chapter “Discussion and conclusions” gives a comprehensive summary of the results and discuss alternative approaches and implications for normalization in biological nervous systems as well as spiking neural networks.

2.2 The biological nervous system

At the heart of any in silico neural network model lies an intent to capture some of the essence of single biological neurons as well as the connectivity and information processing found in biological nervous systems. They are thus all based on the idea of connecting a large number of small independent processing units - with a greater or lesser similarity to biological neurons - into a network where information can be transferred, processed and interpreted. From the first generation of artificial neural networks using highly simplified neurons with binary output, over networks using more realistic phenomenological spiking neuron models, to highly detailed network models, where ion-channel mechanics and the neuronal spatial structure is modelled in detail, the starting point remains the same: A more or less simplified model of a single biological neuron.

Biological neurons

A biological neuron is an electrically excitable cell well adapted to receive process and transmit information (Purves et al. 2008). Neurons vary widely in shape, size and electrochemical properties, but a typical neuron can be divided into three parts: The soma, which contain the cell nucleus, the dendrites which receive most of the incoming connections, and the axon which extends away from the cell body and branches out to form outgoing connections to other neurons. Figure 2.1 shows a schematic picture of a single neuron, as well as some examples of the diverse shapes of biological neurons in the primate brain.

Neurons communicate with each other by sending electrical impulses called action potentials or spikes along their axon. Incoming electrical information is in most cases passively propagated via dendrites and soma while outgoing impulses are actively amplified in the axon to facilitate information transfer over long distances. Between neurons specialized connections – synapses - transmit the signal from the presynaptic cell to the postsynaptic cell by electrical or chemical signaling (Purves et al. 2008). The human brain is comprised of about 100 billion neurons and a single neuron receives input from on average 1000-10 000 other neurons (Herculano-Houzel 2009) which gives the brain a magnitude of 100 trillion synapses. Since a single neuron exhibits a degree of randomness in its behavior the nervous system can be considered a massively parallel, non-deterministic information processing unit.

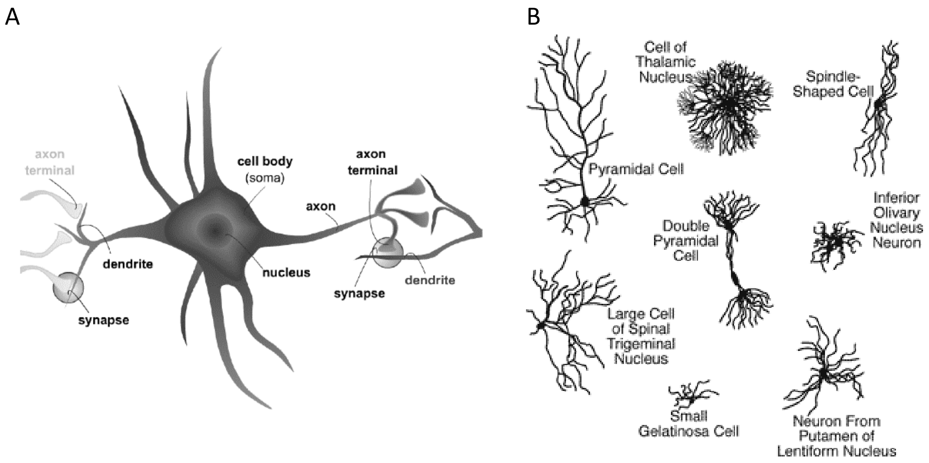

Figure 2.1: A) Schematic drawing of single neuron showing connections with pre- and postsynaptic cells. B) Some of the diverse neurons found in the primate brain. Images from Stufflebeam (2014).

Electrical properties of biological neurons

The electrical potential across the cell membrane seen for all neurons at rest, is the result of a difference in ion concentration between the intracellular and extracellular fluid. This ionic gradient is actively upheld by transmembrane proteins which move ions against their concentration gradient. When no synaptic input reaches the neuron the potential will settle at a value decided by the ionic gradient for the specific neuron type and surrounding temperature. This potential is called the resting potential and varies for different neurons but usually lies around - 65 mV (Purves et al. 2008).

The arrival of synaptic input to a cell can, by the means of opening or closing ion channels, result in a net flow of charge across the cell membrane and thus a change in the potential difference across the membrane. Excitatory neurotransmitters such as glutamate tend to activate ion channels that depolarize (increase) the membrane potential and thus make it more likely for the neuron to fire. Inhibitory neurotransmitters such as GABA on the other hand affect ion channels that tend to hyperpolarize (decrease) the membrane potential and thus make it less likely for the neuron to fire (Purves et al. 2008). The ion channels dynamics are also affected by intrinsic neuronal parameters such as change in membrane voltage or concentration of certain ions.

When input from other neurons cause the membrane potential to rise high enough, an action potential is initiated by the intrinsic dynamics of the neuron, and will propagate along the axon. This will cause release of neurotransmitters at the synaptic terminals which will in turn affect postsynaptic cells. A simplified way of modelling the neuron membrane that capture its basic electrical properties is as a RC-circuit, where the different ion channels are modelled as resistors and voltages in series, and these in turn are placed in parallel with a capacitor representing the membrane capacitance (Steratt et al. 2011). Figure 2.2 shows such an equivalent circuit for a patch of membrane.

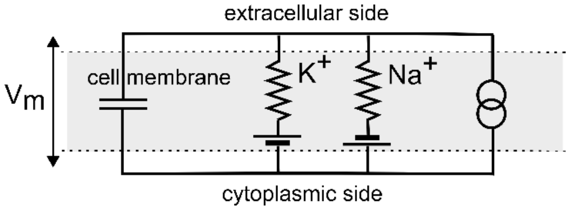

Figure 2.2: Equivalent circuit of patch of passive neuronal membrane. The Na⁺ "battery "tend to depolarize the cell and the K⁺ "battery" to hyperpolarize it. The cell membrane can accumulate charge and thus works as a capacitor. V_m is potential difference across the cell membrane. An electrode by which current can be injected in the cell is also included in the picture. Picture drawn with inspiration from Steratt et al. (2011).

In the circuit represented in Figure 2.2, no dependence of the ionic conductances on the membrane voltage is represented. But this dependence is actually what underpins the formation of an action potential in a biological neuron. When the membrane reaches a threshold potential, voltage dependent sodium channels are opened thus depolarizing the membrane even further, until the slower activation of voltage gated potassium channels again forces the membrane back to the resting level (Purves et al. 2008). The equivalent circuit model can be extended by letting the resistance of the different ion channels change with time and membrane voltage, which is done e. g. for the Hodgkin-Huxley neuron model described in Section 2.3.1 (Steratt et al. 2011).

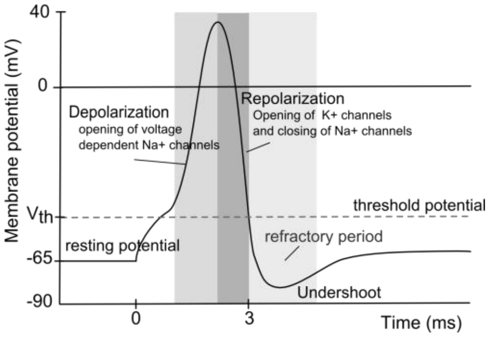

Figure 2.3: Schematic drawing of a generic action potential. At t = 0 a steady state current is injected which causes the membrane potential to rise. When it reaches the threshold potential an action potential is initiated. Drawn with inspiration from Marsland (2009).

The time course of a generic action potential can be seen in Figure 2.3. Note that under a short period after firing an action potential, there is a membrane potential undershoot caused by a delayed potassium current. This makes it very hard to invoke an action potential during this time, which is called the refractory period (Purves et al. 2008). Some neuron models reproduce this behavior intrinsically and for others an absolute refractory period can be added.

For a specific neuron type all action potentials have essentially the same shape. This means that no information is contained *in the shape*, but the action potential can be considered as an all or nothing event (Steratt et al. 2011); it is thus only the firing rate and/or timing of individual spikes that convey information.

If a constant current is injected in a biological neuron, it is possible to get the current - firing frequency relation for a specific neuron. This can be visualized as an input – output curve ("IO curve"). This is also called response curve since it determines the neuronal response to a rate coded input. The terms IO curve and response curve are used interchangeably during this report. The output frequency curve is usually only piecewise linear and saturates for high frequencies. This limits the dynamic range of a biological neuron. It is also important to note that in a biological neuron current is never "injected" but is has to enter the cell through synaptic ion-channels, which can result in different dynamics, e. g. by saturation at the synaptic level (Lansner 2014).

Synapses and neurotransmitters

As with neuron types there are a wide variety of different synapses. The time course over which they affect the permeability of ion channels, the type of chemicals used as neurotransmitters and the amount of synaptic adaptation and facilitation varies greatly (Purves et al. 2008). Synapses are sometimes placed not only on the dendritic tree, but on the soma or even on the axon. Chemical synapses can be classified according to the neurotransmitter released such as glutamatergic or GABAergic. Another important feature is if they have fast ionotropic receptors with a synaptic time constant of a couple of ms or metabotropic receptors that can affect membrane conductance over a time of seconds to minutes (Steratt et al. 2011). There are also electrical synapses where electric current can pass directly between pre- and postsynaptic cells through specialized channels called gap-junctions.

While the form of an action potential is assumed to be the same, the synaptic strength (that is how much charge flowing across the membrane that results from of a presynaptic spike) varies between individual synapses. This also changes dynamically in the living brain in accordance with several different mechanisms such as depression (weakening of a synapse) potentiation (strengthening of a synapse) and facilitation (response becomes stronger if preceded by another action potential) (Purves et al. 2008). Short-term depression and facilitation is shown to influence the information processing of neural circuits (Silver 2010), while long term synaptic plasticity is believed to explain how we can store memories and learn. Because of this significance synaptic plasticity is often also included in neural network models as is the case for one of the models in this thesis.

Cell assemblies and canonical microcircuits

A neuron never exists in a vacuum but is always connected to other neurons. Together they form neural circuits or cell assemblies which is where the truly interesting things happen. These can be seen as functional modules that receives input, process it and then passes the output on to other parts of the brain. Circuits with surprisingly similar characteristics have been found to be repeated across different parts of the nervous system and in different species (Douglas & Martin 2004), which has led to the notion of canonical microcircuits. These functional modules can be reproduced and cascaded to perform complex computational tasks (Carandini & Heeger 2012) in the same way we can construct a highly complex computer from more simple individual parts.

In contrast to today’s computers however, a biological neuron is not a deterministic processing unit. If the same input is repeated several times, the output will most likely not be the same even though in accordance with a certain probability distribution. The randomness in the response of single neurons depends on random fluctuations in the membrane potential generated by noise from other neurons, and the stochastic nature of opening and closing of ion channels (Naud and Gerstner, 2009). This is in line with the idea of neurons or neural assemblies representing information in a probabilistic way.

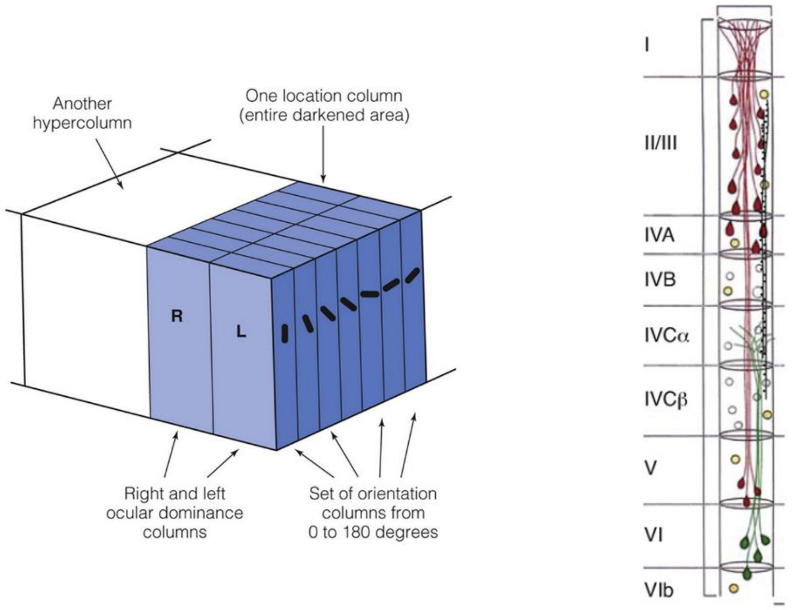
Another hypercolumn
One location column (entire darkened area)
R
L
Right and left ocular dominance columns
Set of orientation columns from 0 to 180 degrees
I
II/III
IVA
IVB
IVCα
IVCβ
V
VI
VIb

Figure 2.4: A) Schematic illustration of the columnar organization of neocortex. B) Illustration of a cortical minicolumn. Letters denote layers and sublayers. Image in A from Cortical structure (2014). Image in B from Lansner (2009).

Cortical structure – Hypercolumns and minicolumns

The mammalian cerebral cortex has a clear laminar structure where different layers harbor different cell types and are believed to have different roles in information processing. The number of layers vary between brain regions, but in most parts of neocortex six different layers can be discerned. Input from thalamus primarily arrives at layer 4, which then projects information to more densely populated layer 2/3 for further processing (Thomson et al. 2002). Apart from the laminar structure of the mammalian cerebral cortex there is also a columnar organization (Hübel & Wiesel 1959). Groups of neurons that share the same thalamic input are called hypercolumns. This means neurons in the same hypercolumn have similar receptive fields while more horizontally distant neurons do not. The columnar organization is also reflected in that vertical connections between neighboring neurons are much more abundant than horizontal ones.

Within a hypercolumn smaller subcolumns called minicolumns can be discerned. Those code for different features within a hypercolumn’s receptive field, such as the orientation of a line as illustrated in Figure 2.4 which show a schematic picture of a hypercolumn. In primate cortex there are 50-100 minicolumns in each hypercolumn and a minicolumn typically consist of around 100 neurons

(Mountcastle 1997). “The columnar organization hypothesis” is currently the dominating explanation of cortical information processing (DeFelipe et al. 2012), but it should be noted that in some mammals, such as rats, the organization shows less evidence of columnar structure.

The hypercolumn module implemented in this thesis is based upon this modular cortical structure. It models primarily layer 2/3 of a neocortical hypercolumn, with input from layer 4 and laterally from other more distant hypercolumns. An important underlying assumption during model construction was that groups of cells – here minicolumns - constitute the fundamental computational unit.

2.3 Models of the nervous system

During the development of the field of computational neuroscience a vast number of different models of neurons and neural systems have been developed, each trying to capture some essence of the biological nervous system information processing and dynamics. This section revisits a number of these and will use the categorization proposed by e. g. Ghosh-Dastidar and Adeli (2009) and Maass (1997) into first, second and third generation of neural networks.

2.3.1 Artificial neural networks

The first generation of neural networks was developed in the 1940s and 1950s, and the basic building blocks of these networks are extremely simplified abstract models of biological neurons. These were named McCulloch and Pitts neurons after those who first presented a mathematical model of a neuron in 1943 (McCulloch & Pitts 1943).

The McCulloch and Pitts neuron (Figure 2.5 A) is a binary threshold device that integrates inputs linearly and fire if the sum exceeds a predefined threshold. T his abstract neuron captures the notion of a neuron receiving input from several sources and firing an action potential if the summed input is sufficiently large. The weights attached to each input, could be interpreted as the strength of the synapse between pre- and postsynaptic neurons. Since the inputs are real valued, they can be seen as representing a rate code from presynaptic neurons (Ghosh-Dastidar & Adeli 2009). The output is however binary; either one, which can be interpreted as that the neuron fires, or zero which would mean it does not.

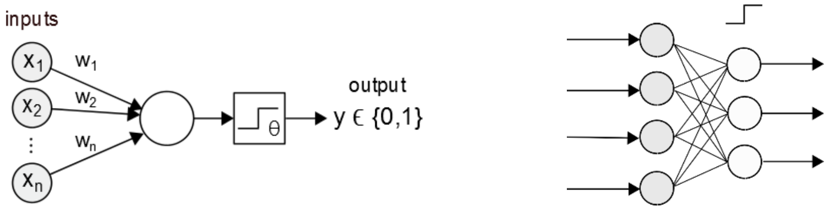

Figure 2.5: A) Schematic illustration of the calculation performed by a McCulloch and Pitts neuron. A weighted sum of the inputs $\sum_i x_i w_i$ is computed and if this exceeds a predefined threshold θ, the output y will be one, otherwise zero.
B) A Perceptron, with three McCulloch and Pitts neurons (white) and four inputs (grey). Each neuron independently performs the calculation visualized in A. Images drawn with inspiration from (Marsland 2009).

Several McCulloch and Pitts neurons can be connected to form an information processing network, referred to as the Perceptron (Figure 2.5 B). For computations with binary output this network is shown to be universal in the sense that it can compute every boolean function (Maass 1997). A single layer network can also be trained by supervised learning to work as a linear classifier. During training the weights are updated according to some learning rule, in analogue with synaptic plasticity in biological nervous system.

To be able to perform non-linear classification the second generation of neural networks was developed which used continuous activation functions such as sigmoidal functions, linear or piecewise linear functions and radial basis functions (Ghosh-Dastidar & Adeli 2009). The basic structure of the network remained the same but the outputs y_j were now described by Equation 2.1.

$$y_j = \varphi\left(\sum_j w_{ij} x_i\right) \tag{2.1}$$

Where x_i is input i, φ the continuous activation function and w_{ij} the weight between input i and neuron j. The real valued output from these units can be interpreted as a rate code. As well as solving non-linear classification problems, a multi-layer perceptron with sigmoidal activation function can reproduce any continuous function with arbitrary precision if enough hidden neurons are used (Cybenko 1989).

Artificial neural networks have since been developed and elaborated into a large number of different architectures such as multilayer neural networks, radial basis function networks, self-organizing maps and dynamic neural networks (Figure 2.6). They have proven useful for a variety of computational tasks such as classification, pattern recognition, function/process estimation, complicated optimization problems and data compression (Marsland 2009).

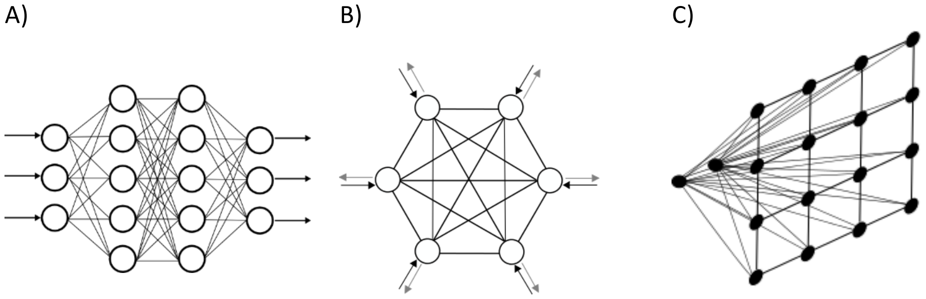

Figure 2.6: Some different types of artificial neural networks. A) Multilayer perceptron. B) Hopfield network. C) Self organizing map.

Important lessons and insights into biological neural systems have resulted from studying artificial neural networks. One example is associative memory models which were first proposed in an artificial neural network context, and has later been implemented with increasing level of biological realism. One such implementation is the spiking neural network attractor memory model introducing the hyper- and minicolumnar organization used for the models in this thesis (Lansner 2009). When drawing analogies between artificial neural networks and the biological neural systems it is thus important to remember that an artificial neuron can be interpreted not only as a single neuron but also as a local subpopulation of neurons (Lansner 2009).

Limitations

The “non-spiking units” of the first and second generation of neural networks are still very far from biological neurons. The most important limitations are that all computation occurs at discrete time steps; all inputs arrive simultaneously; the summation of inputs is always linear; and the firing threshold of a neuron is fixed. They can thus not represent the continuous temporal dimension of real neurons or more complex intrinsic neuronal dynamics. Usually there is also no connection to biological processes for learning or only a pseudo-realistic learning mechanism (Ghosh-Dastidar & Adeli 2009). Thus although useful for a vast number of computational tasks and able to represent some neural network phenomena conceptually these limitations makes those artificial neurons less useful for understanding or reproducing more complex biophysical phenomena.

To overcome these limitations model neurons that can implement more biologically realistic and diverse biophysical dynamics as well as encode information via the precise timing of spikes have been developed. Those are the basis for the third generation of neural networks - spiking neural networks (Ghosh-Dastidar & Adeli 2009).

2.3.2 Spiking neural networks

Growing experimental support of the possible importance of spike timing for neural computations (Ghosh-Dastidar & Adeli 2009) and processing speed, as well as robust long range communication in neural systems (Ponulak & Kasinski 2011) were some motivations for the development of spiking neural network models. Models comprised of spiking neurons can also more accurately take into account the continuous temporal dimension, and local nature of biological neural systems.

The more complex behavior of spiking neurons can result in more computational power: When a neuron have more complex behavior the information or computations possible to represent with the same number of neurons increase (Maass 1997). Although it we still do not know for sure if and how this is actually utilized by biological nervous systems, spiking neural networks are widely used to gain increased understanding into the workings of biological nervous systems. They have also been proven useful for biomimetic algorithms that enable fast information processing and learning such as fast signal-processing, classification, speech-recognition and event detection (Ponulak & Kasinski 2011).

Spiking neurons

In spiking neural networks the computational unit is a spiking neuron which is not updated at specific discrete points in time, but has a dynamic internal state described by at least one state variable. For more complex models with several state variables the spiking neuron will be a small dynamical system of coupled differential equations. The parameters in spiking neuron models are usually related to actual physical quantities such as membrane voltage, capacitance and permeability.

Similar to the ANN neuron a spiking neuron will fire when given enough input. However the input and membrane voltage for which this happens may vary. Similar to biological neurons the magnitude of a spike is assumed to contain no information; the firing of an action potential is an all or none response. Since the precise timing of spikes and spike trains can be modelled, a spiking unit can in contrast with an ANN neuron use not only rate but also temporal encoding (see

Section 3.1). Spiking neuron models can also reproduce complex biological firing patterns such as bursting and slow or fast adaptation and facilitation.

2.3.3 Designing spiking neural networks

When modeling a spiking neural network, as we do in this thesis, maybe the hardest choice to make is how detailed it should be, since for each level of detail it is always possible to make an even more detailed model. We do not yet know how much of the astonishing detail present in biological nervous systems that is necessary for its information processing purposes; it is thus often unknown how much biological realism that is needed for the specific task at hand. In addition, biological data for parameters such as connectivity between different neuron types as well as for neuronal and synaptic parameters are often incomplete or uncertain. This means when constructing a spiking neural network model it is necessary to make an informed guess of the features and complexity needed.

Previously the computational cost for large realistic network models often constrained the level of complexity (Ghosh-Dastidar & Adeli 2009). With the constant increase in computational power this is no longer the case, even if something more powerful than a standard laptop might have to be used for simulations. However, that it is possible to include more details does not always mean it is best to do so. It can also make the underlying mechanisms more unclear - and usually those are what we wish to reveal - and the model and processes harder to understand (Steratt et al. 2011). In order to relate to higher functionalities such as computation or coding, it e. g might not be necessary to model details of ionic flow and protein interactions. And for real-time applications such as small embedded systems, computational complexity can still be an issue. All in all this means a complicated model needs to be justified just as well as a simple one.

As for many other scientific undertakings, it is probably good to keep in mind the quote with somewhat unclear origin but often attributed to Einstein:

“Everything should be made as simple as possible but not simpler”

Following this advice, the models implemented in this thesis are not using more complex features than can be justified by their functionality in relation normalization. It is however necessary to accept that each study will explore only a certain part of a vast “complexity space”, and even though we of course try to choose the part believed to be most enlightening there are no guarantees.

2.4 Spiking neuron models

The multitude of different models describing the dynamics of a spiking neuron, all try to capture some of the essence of a biological neuron. In common for all spiking neuron models is a dynamic state variable corresponding to the neuron membrane potential. This variable is updated continuously in correspondence with incoming presynaptic events and intrinsic phenomena such as leak current through the membrane or the generation of an action potential. When a certain internal state is reached, a spike will be emitted and propagated to postsynaptic neurons.

This said the different neuron models vary greatly in biological realism and can thus capture different aspects of - and in more or less detail - the extremely diverse properties and behavior of biological neurons. Hodgkin-Huxley like models have more of a bottom-up approach (Ghosh-Dastidar & Adeli 2009). By simulating the differential equations governing the membrane conductances resulting from opening and closing of ion-channels and the spreading of post synaptic potentials and action potentials in different parts of the neuron, the models can be calibrated with biological data and reproduce complex behaviors found in biological neurons (Steratt et al. 2011). Hodgkin Huxley multi-compartment models of single neurons can have more than thousand compartments, coupled to each other by boundary conditions of their respective differential equations. Multi-compartmental models can thus in great detail model neurons of different spatial structure and how e. g. placement of synapses or branching of dendrites affect neuronal properties.

The less detailed models have a functional or top-down approach where the parameters in the model are chosen not to necessarily represent actual biological quantities but to reproduce specific behavior or firing characteristics of biological neurons. Integrate-and-fire, Izhikevich and Morris-Lecar neuron models belong to this category. These are also all point neuron models, which means the spatial structure of the neuron is not taken into account but it is modelled as a single compartment. These simpler models can often allow mathematical analysis of a network, e. g. via dynamic system analysis (Steratt et al. 2011). The behavior of networks composed of more complex neurons can on the other hand only be studied experimentally.

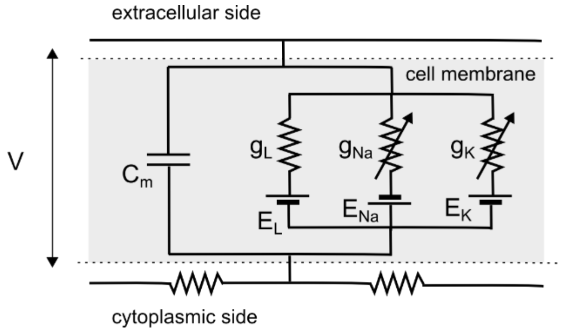

Figure 2.7: Equivalent circuit for the Hodgkin-Huxley neuron model. The arrows in the Na and K resistors show that they are active voltage dependent conductances. Drawing inspired from Steratt et al. (2011).

2.4.1 Hodgkin-Huxley

The Hodgkin-Huxley neuron model (Figure 2.7) was the first quantitative model of active membrane conductances. It was developed by Hodgkin and Huxley in the early 1950s to describe the initiation and propagation of the action potential in a giant squid axon (Hodgkin & Huxley 1952). In the basic version the Hodgkin-Huxley model includes three ionic currents: sodium, potassium and leak current and the current through the membrane is taken to be the sum of these currents. The leak conductance is further assumed to be independent of the membrane voltage, while the sodium and potassium currents are modeled as active, that is depending on both voltage and time. The ion channels are modelled as dependent on different voltage and time dependent “gating particles” which regulate the activation (opening) and for potassium current, also inactivation (closing after a certain time at a high membrane voltage) of the ion channels (Steratt et al. 2011). These gating particles are a theoretical construct without any direct biological counterpart. The molecular mechanisms underlying the ion-channel dynamics are thus not modelled directly. The result is four coupled differential equations (Equation 2.2) which model as well the neuron subthreshold dynamics as the duration, amplitude and shape of an action potential (Hodgkin & Huxley 1952).

$$C_m \frac{\partial V}{\partial t} = -\, g_L(V - E_{L)} - \bar{g}_{Na} m^3 h(V - E_{Na)} - \\ \mathrm{g}_K n^4(V - E_{K)} + \frac{d}{4R_a}\frac{\partial^2 V}{\partial x^2} \tag{2.2}$$

$$\frac{dm}{dt} = \alpha_m(1 - m) - \ \beta_m m$$

$$\frac{dh}{dt} = \alpha_h(1 - h) - \ \beta_h h$$

$$\frac{dn}{dt} = \alpha_n(1 - n) - \ \beta_n n$$

Here C_m is membrane potential, g_X, maximum value of the respective conductances, E_x reversal potential for leak, Na, and K current respectively m, n and h the “gating particles” and α and β experimentally fitted rate constants corresponding to transition rates between the “open” and “closed” state of the gating particles. Note that while the K^+ channel only has an activation variable n, the Na^+ channel also has an inactivation variable h, resulting in decreased conductance after a certain time at a high voltage. This is what causes the decay of the membrane potential back to resting potential. A refractory period is reproduced by this model since the recovery of the gating variables is considerably slower than the time course of the action potential (Steratt et al. 2011). The Hodgkin-Huxley model can account for the variable spike threshold and refractory period seen in biological neurons, as well as different spike characteristics. It is often used in multi-compartmental models.

Multi-compartmental models

To account for the three dimensional structure of biological neurons and current flowing not only across the membrane but also perpendicular to it, it is necessary to model how current and voltage is transmitted between different parts of the neuron. This is usually done by dividing the neuron into a number of compartments each governed by for example a Hodgkin-Huxley type equation. The compartments are then connected by allowing inflow of current from neighboring compartments. Compartmental models can be used to account for behavior of biological neurons not possible to capture with a single compartment model such as different effects from inhibition depending on where on the dendritic tree inhibitory synapses are placed; presynaptic inhibition; and non-linear processes in the dendrites.

2.4.2 Morris-Lecar

Several simplifications of neuronal dynamics to a model with only two state variables have been proposed after the Hodgkin-Huxley model was constructed (Steratt et al. 2011). The two state variables are usually a faster voltage variable and a slower recovery variable. In the Morris-Lecar model the reduction is done by assuming that the dynamics of the neuronal membrane is described by only K^+ and Ca^+ ionic currents. It is also assumed that the calcium current respond instantaneously to voltage. This gives two state variables: membrane potential and the potassium state. The system can thus be governed by only two coupled differential equations:

$$C_m \frac{dV}{dt} = -I_{ion}(V, w) + I_e \tag{2.3}$$

$$\frac{dw}{dt} = \Theta \cdot \frac{w_\infty(V) - w}{\tau_w(V)}$$

$$I_{ion}(V, w) = g_{Ca} m_{inf}(V - V_{Ca}) + g_K * w(V - V_k) + g_L(V - V_L)$$

$$m_\infty(V) = 0.5(1 + \tanh(\,(V - V_1)/V_2))$$

$$v_\infty(V) = 0.5(1 + \tanh(\,(V - V_3)/V_4))$$

$$\tau_w = \phi/\cosh((V - V_3)/(2V_4))$$

Here I_e is injected current, I_{ion} ionic current, m_{inf} Ca^{2+} activation variable, w K^+ activation variable, w_∞ steady state K^+ activation variable, tau_w K^+ activation time constant, and ϕ represents a temperature/time scaling. The parameters V1, V2, V3 and V4 determine the activation curves for the calcium and potassium voltages (Steratt et al. 2011).

With only two time-dependent variables V and w, this model is computationally much cheaper than Hodgkin-Huxley while still able model intrinsically the generation of an action potential. The Morris-Lecar model can produce different IO curves depending on the choice of parameters (Steratt et al. 2011) and also model spontaneous firing rhythm patterns, such as bursting.

2.4.3 Izhikevich

Another two state variable neuronal model is the Izhikevich model. This model can reproduce a wide range of behaviors seen in biological neurons using two quite simple coupled ordinary differential equations (Izhikevich 2003). Examples of firing patterns are shown in Figure 2.8. The state variables for the Izhikevich model is as for the Morris-Lecar model a fast voltage variable v, and a slower recovery variable u. There is no longer any clear correspondence between the model variables and biological quantities, but the constants a and b in the model are chosen to reproduce specific firing characteristics. The Izhikevich neuron model is described by Equation 2.4.

$$\frac{dV}{dt} = k(V - E_m)(V - V_{\theta)} - u + I \tag{2.4}$$

$$\frac{du}{dt} = \; a(b(V - E_{m)} - u)$$

$$if\ V\ \geq 30\ mV, then \begin{Bmatrix} Reset\ V\ to\ c \\ Reset\ u\ to\ u + d \end{Bmatrix}$$

The equations can model the *rise* of the action potential, but not the fall back to resting potential. This is instead done by resetting when membrane potential reaches a predefined value. The model does not necessarily have a fixed firing threshold and is in this aspect similar to the Hodgkin-Huxley model. The relative simple governing equations enable simulating very large networks with moderate computational resources.

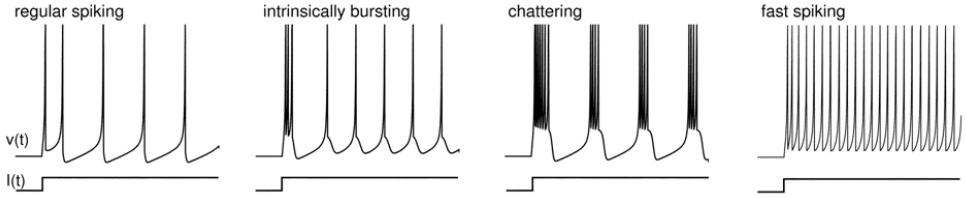

Figure 2.8: Some of the diverse firing patterns exhibited by Izhikevich neurons when injected by a constant current I(t). Vertical axis show membrane voltage and horizontal axis time. Picture from Izhikevich (2003).

2.4.4 Integrate-and-fire

The integrate-and-fire (IAF) neuron model is a further simplification of neuronal dynamics to just one state variable: membrane voltage. The subthreshold neuronal dynamics are be described by one linear differential equation and the action potential is simply modelled as a discrete event. When the membrane potential reaches a predefined threshold V_{th}, a spike occurs and the membrane voltage is reset to a reset potential V_r and clamped there during the refractory period τ_r.

$$C_m \frac{dV}{dt} = -\frac{V - E_m}{R_m} + I \tag{2.5}$$

$$if\ V = \ V_{th}\ then\ emit\ spike$$

$$and\ clamp\ V\ to\ V_R\ during\ \tau_r\ seconds$$

Here C_m is membrane conductance, R_m membrane resistance, E_m membrane resting potential and I injected current or current resulting from synaptic input. The neuronal dynamics are thus treated as those of the simple RC-circuit in Figure 2.2. The leak current - described by the first term on the right hand of the equation - is included to account for that the neuronal membrane is not a perfect isolator. Often the resting membrane potential E_m is set to zero and the equation is expressed in terms of the membrane time constant, $\tau_m = C_mR_m$. This is a convention followed also in this thesis and gives Equation 2.6.

$$\tau_m \frac{dV}{dt} = -V + IR_m \tag{2.6}$$

The simple dynamics of the IAF neuron model makes it straight forward to analyze analytically. Solving Equation 2.6 for the case of a small constant injected current I, e. g gives an expression for how the membrane potential evolves with time:

$$V = R_m I(1 - \exp(-\,t/\tau_m)) \tag{2.7}$$

As can be seen, the membrane potential approaches R_mI asymptotically and thus the neuron will only fire if R_mI is bigger than V_{th} otherwise the membrane voltage will saturate at a constant elevated level. For input current large enough for the neuron to reach threshold, the steady state firing rate for different amounts of constant input current is described by Equation 2.8; which thus describe the input-output (IO) curve of the model neuron.

$$f(I) = \frac{1}{\tau_r + \tau_m \ln(1 - V_{th}/R_m I)} \tag{2.8}$$

If no refractory period is included the firing frequency will increase with input without bound, but otherwise it will asymptotically approach the limit given by the refractory period. The firing characteristics of an IAF neuron are illustrated in Figure 2.9.

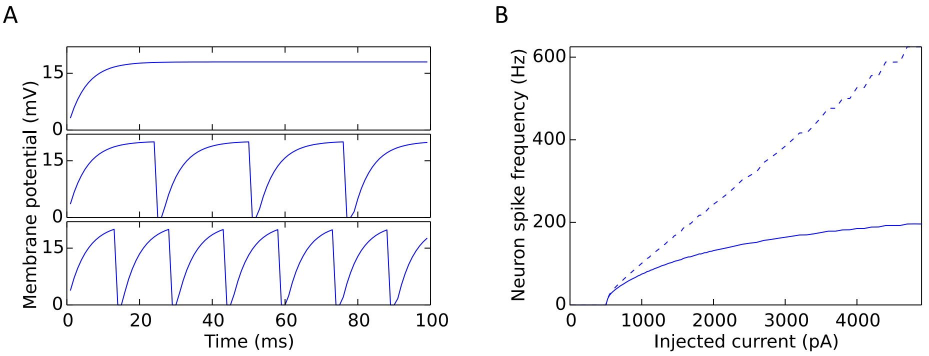

Figure 2.9: Firing characteristics of the IAF neuron model. A) Response of IAF neuron to different amounts of constant input current, chosen to result in a membrane potential rising precisely above or below firing threshold. B) IO curve for IAF neuron. Continuous line – with refractory period. Dashed line - without refractory period. Figures created with inspiration from Steratt et al. (2011).

The low computational complexity of the IAF neuron model enable fast simulation of very large networks of neurons. It also enables mathematical analysis of large networks, for example by using mean field theory (Brunel 2000). The direct representation of membrane voltage makes it possible to model how e. g. noise affects membrane variability and consecutively the response curve, which is not possible using simple rate based units. The basic IAF model can however not reproduce more complex neuronal firing patterns such as bursting, intrinsic adaptation or variable firing threshold.

Adaptive integrate-and-fire neurons

It is possible to add adaptation to the IAF neuron model by including a negative adaptation current that grows larger for each initiated spike or in proportion to an elevated membrane potential. During repeated firing this makes it harder and harder to initiate a spike. One simple model shown to be able to reproduce adapting spike patterns similar to biological neurons, was developed by Brette and Gerstner (2005). The subthreshold membrane potential is described by Equation 2.9, and the adaptation current by Equation 2.10. An exponential term is also added in the equation describing subthreshold dynamics.

$$\begin{gathered} C_m \frac{dV}{dt} = -g_L(V - E_L) + g_L \Delta_T \exp\left(\frac{V - V_{T\theta}}{\Delta_T}\right) - \omega + I \\ \tau_w \frac{dV}{dt} = a(V - E_L) - w \\ if\ spike\colon\ w \rightarrow w + b \end{gathered} \tag{2.9}$$

Where τ_w is the time constant for adaptation, a represents the level of subthreshold adaptation and b the amount of spike-triggered adaptation. V_T is a threshold voltage and Δ_T a slope factor that determines the sharpness of spike initiation. This model is used to include adaptation for model neurons in one of the model versions implemented in this thesis.

2.4.5 Synaptic input to spiking neurons

Synaptic input to a neuron can be “plugged in” to the governing equations of a spiking neuron as the current resulting from synaptic events. For a conductance based synapse this would mean the synaptic current represented by I in Equation 2.6 would be described as:

$$I_{syn}(t) = \sum_j g_{syn\_j}\left(V(t) - E_{syn\_j}\right) \tag{2.10}$$

Where the sum is over all incoming synaptic connections and E_{syn} is the equilibrium potential for each activated synapse. The function $g_{syn}(t)$ describes the time course of the membrane conductance change and can be approximated in different ways. One common way, shown to be a quite realistic representation of the synaptic input to a biological neuron, is to model it as an alpha function:

$$g_{syn}(t) = \begin{Bmatrix} g_{max}\dfrac{(t-t_s)}{\tau_{syn}}\ \exp(-\dfrac{(t-t_{s)}}{\tau_{syn}}) & for\ t \geq t_s \\ 0 & for\ t < t_s \end{Bmatrix} \tag{2.11}$$

Here t_s is the time when the synapse is activated, g_{max} the maximum conductance and τ_{syn} the synaptic time constant. A larger τ_{syn} means that conductance changes affect the neuron during a longer time period. A simplification is to instead use a decaying exponential function with instantaneous rise time:

$$g_{syn}(t) = \begin{Bmatrix} \bar{g}_{syn}exp\left(-\dfrac{(t-t_s)}{\tau_{syn}}\right) & for\ t \geq t_s \\ 0 & for\ t < t_s \end{Bmatrix} \tag{2.12}$$

Other more complex functions describing the synaptic input exists but are not considered here. A further simplification which facilitates mathematical analysis and enables faster simulation is to instead use current based synapses. For current based synapses the time course of the synaptic current is modelled directly as in Equation 2.14.

$$I_{syn}(t) = \begin{Bmatrix} \bar{I}_{syn}exp\left(-\dfrac{(t-t_s)}{\tau_{syn}}\right) & for\ t \geq t_s \\ 0 & for\ t < t_s \end{Bmatrix} \tag{2.13}$$

This could of course also be done with an alpha shaped postsynaptic current. For all model versions in this thesis exponential shaped conductance based synapses are used.

Synaptic plasticity

To capture more realistic neuronal behavior with an IAF neuron it is also possible to include synaptic plasticity in the form of long or short-term facilitation or depression. Long-term synaptic depression has been found to have a large impact on biological neural systems; these semi-permanent changes in synaptic strengths makes it possible for the brain to learn and adapt to new conditions. Long term depression or potentiation is usually modelled as an increase/decrease in synaptic weights. Short-term depression on the other hand can be important for information processing on a much smaller time scale; the effect usually wears off after a few hundred milliseconds. One quite simple model that describes short-term synaptic depression phenomenologically as a consequence of depletion of synaptic resources was developed by Tsodyks et al. (1998). They developed a model of synaptic depression by considering the three states a synaptic neurotransmitter can be found in: recovered (ready for release), effective (in synaptic cleft), and inactive (no longer active and not yet but soon, to be recovered). Each of those states are assigned a state variable and their dynamics are modelled as a system of differential equations:

$$\begin{aligned} \frac{dx}{dt} &= -\frac{z}{\tau_{rec}} - u\,\delta(t - t_{sp}) \\ \frac{dy}{dt} &= -\frac{y}{\tau_{in}} + u\delta(t - t_{sp}) \\ \frac{dz}{dt} &= \frac{y}{\tau_{in}} - \frac{z}{\tau_{rec}} \end{aligned} \tag{2.14}$$

Here x represents the recovered, y the effective and z the inactive part of the neurotransmitter and u [0,1] the release probability. For each incoming spike thus u percent of the recovered neurotransmitters is released and enters the effective state. The net synaptic current/conductance will be proportional to the fraction of resources in this state. The recovered pool will thus diminish for each spike while the active pool increases. This is the reason the effect a single synaptic event decreases for a sequence of consecutive spikes: if all synaptic resources are in the active pool, there are none left to release. The conductance change will thus be a saturating function of input frequency. The inactive pool on the other hand grows in proportion to the active neurotransmitters and decays back to zero at as the neurotransmitter enter the recovered stage again. Thus after a “resting period” the synapse will again be ready to release its maximum amount of neurotransmitter. This is the model used when synaptic depression is included in the implemented model versions.

Modelling short-term facilitation can be done by letting also u be a dynamic variable changed for each incoming spike by incrementally increasing the release probability, u, (Tsodyks et al. 1998). The dynamics are described by Equation 2.15:

$$\frac{du}{dt} = -\frac{u}{\tau_{facil}} + U_{SE}(1 - u)\delta(t - t_{sp}) \tag{2.15}$$

Here U_{SE} represent the percentage increase in u and τ_{facil} the decay time. Synaptic facilitation is not included in any of the network model versions.

3 Theory

3.1 Neural coding

Somehow the neural activity in the brain encodes information to enable diverse and complex tasks such as processing multisensory information or storing and retrieving information on different time scales. In a digital computer information is both stored and communicated in binary code. We know exactly how it works since we ourselves designed the system. In the brain on the other hand, although we have a quite good idea about the neural code for *some tasks*, for most we do not yet fully understand how information is encoded in communication between neurons or stored as physical changes in the brain. This is a serious limitation when trying to understand biological neural circuits and poses an additional difficulty when designing biomimetic neural networks. As stated by Louie et al. (2013)*:*

> *"Understanding the neural code is critical to linking brain and behavior"*

It is indeed also critical to designing biologically realistic neural networks. But a different approach can be to design a network that is similar in structure to some biological neural circuit and then see what type of coding regime this network can utilize. This might help solving the puzzle. Since the brain is a highly diverse biological device, information coding is most likely not as simple and clear cut as for computers; most probably different means of encoding are used for different functions as well as for different species and brain regions.

Rate coding vs temporal coding

Information transmitted from pre- to postsynaptic neurons is often characterized as using one of two coding regimes: rate coding or temporal coding. There is no complete consensus of the definitions, but usually the term "rate coding" is used to describe a neural code where only the average firing of a neuron or group of neuron carries information, while the term "temporal coding" is used for proposed coding regimes where also the precise timing of individual spikes is believed to be important (Kostal et al. 2007). Temporal coding can e. g. mean that information is transmitted by the correlation of spikes (Silver 2010) or is contained in the time to first spike or the temporal structure of the interspike intervals (Kostal et al. 2007). Since there is no global clock in the nervous system, a temporal code is somehow based on *relative* timing of spikes, e. g. in relation to an external stimulus or global oscillations in a brain area.

Rate coding as a means of representing a continuous sensory variable such as touch or pressure was first described in sensory neurons by Adrian and Zotterman (1926). They showed that sensory neurons responded with regular sustained firing of a frequency proportional to stimulus intensity. For rate coded information a computational unit's response curve is a good way of describing its information processing.

For some brain regions, such as neocortex, where the firing rate of individual neurons can be as low as a few Hz it has been argued that rate coding would be unfeasible since it would simply take too long to get a reliable average (Silver, 2010). However, if the computational unit is not a single neuron but a group of neurons - such as a cortical minicolumn - this is does no longer have to be a problem. The time needed to get an average of the firing for a *group of neurons* is considerably shorter, and could thus very well be compatible with the response times seen in cortex (Shadlen & Newsome 1998). Population rate coding could also be a means for neural systems to represent probability distributions and thus explain the near optimal probabilistic inference found in psychophysical experiments (Ma et al. 2006). Population rate coding is the coding regime assumed for the models in this thesis. It should be mentioned that also if primarily firing rate is studied the use of spiking neurons can be motivated because of their more complex behavior compared rate based units.

Local vs distributed coding

Another dimension to consider for neural coding is spatial distribution of active neurons. In one end of the spectra lies a local code where every neuron code for one single feature and in the other a dense code where all neurons in a population take part in representing all possible features. A compromise between these extremes is a sparse code where a small percentage of neurons in a population is active for a certain stimulus (Silver 2010). It has been shown theoretically that sparse coding is the most energy efficient way of representing information and should thus be favored evolutionary (Olshausen & Field 2004). There are also evidence of sparse coding in primate cortex (Waydo et al. 2006). For cortical hypercolumn such as the one modelled in this thesis, it has been shown only a few of the minicolumns in a cortical hypercolumn are active simultaneously.

3.2 Non-linear cortical operations

The classical view of neurons is that they compute a linear weighted sum of their inputs and answer with a firing rate, which is a function of this sum. If this were true the complex computations implemented by neural networks would have to be the result of complex connectivity and not attributed to the computational power of individual neurons (Silver 2010). But the weighted linear summation of inputs, have been shown to be only part of the story: Complex as well as morphologically simple neurons seem to be able to perform a range of arithmetic operations such as division and exponentiation (Silver 2010).

Through this thesis the division of neural input into driving and modulatory input introduced by Sherman and Guillery (1998) is used. While they put forward a possible anatomical distinction between the two (where drivers were proposed to act through fast ionotropic receptors and modulators activate slower metabotropic receptors) this could also be more of a functional distinction as suggested by Abbott and Chance (2005). Driving input might e. g. represent the strength of a sensory input, and the modulatory input how some other variable (such as attention, gaze direction or other concurrent stimuli) modifies the neural response.

Addition and subtraction by neurons

Neuronal firing indeed typically represent the sum of synaptic inputs (Rothman et al. 2009). This computational mode means that if a modulatory drive, m, is added to the driving input the result is to change the output of the neuron from $R(d) = f(d)$ to $R(d, m) = f(d + m)$. The response curve is thus shifted to the right or left on the horizontal axis, depending on the sign of the modulatory input. That adding a certain amount of excitatory or inhibitory current to a neuron give this effect is easily motivated by considering the simple RC-circuit model of a neuron; all positive and negative currents are summed up and thus affect the membrane potential as well as the firing rate, in an additive way.

A textbook example of this type of calculation is the explanation of how linear summation of afferent inputs give rise to the direction selectivity in simple cells in primary visual cortex (Hübel & Wiesel 1959).

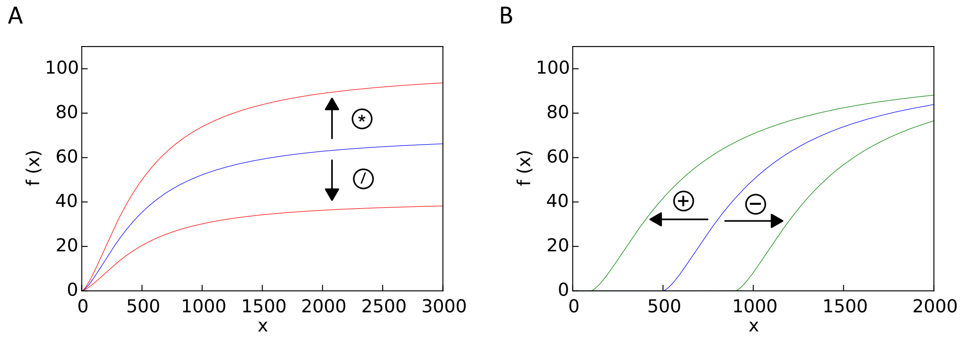

Figure 3.1: Schematic picture of multiplicative and additive input modulation. A) Multiplicative/divisive input modulation, according to the "output gain" model. This affects the slope of IO curve as well as max output. B) Additive/subtractive input modulation shifts the IO curve on the horizontal axis. Pictures drawn with inspiration from Silver (2010).

Multiplicative operations

However, there are evidence a neuron can also perform multiplicative operations (Silver 2010). A multiplicative modulatory input m, would transform the output of a neuron from R(d) = f(d) to R(d, m) = f(d*m) or alternatively R(d, m) = m*f(d). The former will be referred to as "input gain control" (the input/ independent variable is scaled multiplicatively) and the latter as "output gain control" (the output/dependent variable is scaled multiplicatively). For both these versions the sensitivity of the neuron to driving input, which will be referred to as "gain", is scaled up or down. It should be noted that normalization is a form of input-gain control. Figure 3.1 A shows the effect on the IO curve from output gain control.

Evidence of multiplicative neural computations are found e. g. during contrast invariance of orientation tuning, attentional scaling, translation-invariant object recognition, auditory processing and coordinate transformations. The necessity of gain modulation for these tasks is highlighted by theoretical studies (Rothman et al. 2009). A review of the possible biophysical mechanisms that can implement multiplication on the neuronal level is given in Section 3.4, but it is important to note the qualitative difference between an additive and multiplicative modulation of the IO curve.

Exponentiation

Another non-linear operation that can be performed by biological neurons is exponentiation. Neuron firing rate with respect to membrane voltage, can typically be described by a threshold function - that is an IO curve with a discontinuous firing onset - but in the presence of synaptic noise this relationship will instead be transformed to a sigmoidal function. This follows since increased membrane variability smoothens the threshold making the neuron sometimes fire also for membrane voltage below the threshold. This feature is also reproduced in biophysically detailed neuron models (Gabbiani et al. 2004), as well as the simple IAF model used for the models in this thesis (see Results chapter). The smoothed firing onset will make the neuronal response to excitatory input approximate a power law of exponential function at firing onset. An exponential response function can indeed be a possible way for neurons to implement multiplication since $\exp(x - b) = \exp(\mathrm{x}) / \exp(\mathrm{b})$ (Silver 2010). However the approximate exponential relationship between membrane voltage and firing rate typically only holds for low input rates, which limits the range of inputs that can be modulated multiplicatively.

3.3 Normalization in biological nervous systems

Evidence of normalization, where the activity of a single neuron/computational unit is rescaled divisively in relation to the activity in a group of other neurons/computational units is found in an increasing number of modalities, tasks, brain regions and different species (Carandini & Heeger 2012). Normalization has been shown to be important to a vast number of brain functions, not only regarding sensory processing where the concept was first developed. Such prevalent neural computations are considered to be "canonical neural computations" and normalization is proposed to be one of those (Carandini & Heeger 2012).

The linear model - where the response of a neuron depends on a weighed sum of intensity values or afferent neural activities - can account for multiple sensory processing phenomena such as edge detection as well as orientation, position and motion selectivity in the visual system (Carandini & Heeger 1994). However, as pointed out by Carandini and Heeger this model fails to account for other phenomena such as cross-orientation inhibition and surround suppression, where the response of a neuron is suppressed by stimuli that has no effect at all if presented alone. Further the linear model cannot explain why for a non-preferred stimulus a doubled stimulus intensity does not give a doubled response, but instead there is saturation at a lower level than the maximum firing rate for a preferred stimulus. To account for these phenomena Carandini and Heeger (1994) proposed a "normalization model" where (in line with the distinction later formalized by (Sherman & Guillery 1998)) the inputs to a neuron were divided into driving and modulatory inputs. The effect of this distinction was that the driving inputs could still be assumed to be summed linearly - and thus account for the linear phenomena - while the modulatory input rescaled the response in a divisive manner. The modulatory input originating from a larger pool of neurons which was termed "the normalization pool".

3.3.1 The normalization model

The normalization model describes how the activity in a neuron or computational unit is rescaled with respect to the input to/activity in a pool of other neurons/ computational units. This gives a response which preserves information about the *relation* between different inputs/stimuli information but is partly or completely invariant to the absolute input magnitude. Normalization in neural systems can be compared with normalization of a vector in a Euclidean vector space, where every entry in the vector is divided by the vector norm resulting in a new vector of norm one. Consider e. g. the p-norm $\hat{\mathrm{x}}$ of an input vector $\bar{x}$ where x_i represents the individual vector entries:

$$\hat{x} = \frac{\bar{x}}{\sqrt[p]{\sum |x|_i^p}} \tag{3.1}$$

Note how every entry is rescaled by the same factor which is also true for the equation describing normalization in biological nervous systems (Equation 3.2). Equation 3.2 can in fact be seen as a continuous version of Equation 3.1, capturing the gradual onset of normalization as well as the saturation present in biological neural systems. The “normalization equation” thus describes how the response of a single neuron or group of neurons, R_j, does not only depend on its driving input Dj, but also on input to other neurons, D_k, in “the normalization pool”:

$$R_j = R_{max} \frac{D_j^n}{\sigma^n + \sum_{j \neq k} D_k^n + D_j^n} \tag{3.2}$$

Here R_{max} determines max response; σ for which input value half the max response is reached without any normalization input; and n the shape of the response curve/amplification of individual signals. Note how increased input to neurons in the normalization pool will have the same effect as increasing σ. The constants are free parameters which are derived experimentally and vary between different neural systems. For higher values of n the model goes towards winner-take-all (WTA) behavior. Two examples of experimental studies that show evidence of normalization in the primate visual system are seen in Figure 3.2.

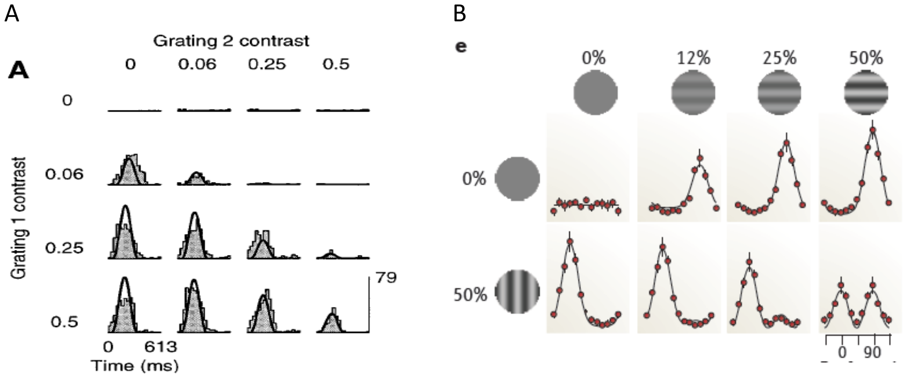

Figure 3.2: Normalization in primary visual cortex (V1). A) Cross-orientation inhibition. Grating 1 has an optimal direction for driving the cell, while grating 2 is an orthogonal grating, unable to drive the cell when presented alone. Note how nonetheless the response to grating 1 is suppressed for higher contrast of grating 2. B) Normalization predicts soft WTA behavior for population responses. Dots represent the response of a neuronal population with a specific orientation selectivity. Lines are fit to the normalization model. Image in A from Carandini et al. (1997). Image in B from Carandini and Heeger (1994).

If the driving input in the normalization model comes from a summation field the stimulus drive will be the weighted sum of sensory inputs as in Equation 3.3:

$$D_k = \sum_k w_{jk} I_k \tag{3.3}$$

In a corresponding way a suppressive field can be defined for the normalization pool inputs defined by weights α_{jk}. These fields can be different for different neurons. A spontaneous driving input β can also be added to the normalization equation - this would correspond to letting also the spontaneous activity be affected by normalization - and different exponents (m, n and p) can be used for the numerator and denominator (Carandini & Heeger 2012). This gives the more general model described by Equation 3.4:

$$R_j = \gamma \frac{\left(\sum_k w_{jk} I_k\right)^n + \beta}{\sigma^n + \left(\sum_k \alpha_{jk} I_k^m\right)^p} \tag{3.4}$$

Evaluating normalization in different neural systems is often done by to fitting the normalization equation to the neuronal responses for different driving and normalizing/modulatory inputs (Carandini & Heeger 2012). Often the exponents in nominator and denominator are assumed equal, and have e. g. in the visual system been found to typically lie between one and four.

That the response of a cell or computational unit in relation to stimuli is accurately described by the normalization model, does not necessarily give away anything about the mechanistic implementation of normalization. Also, since normalization is often studied directly as one neurons response to external stimuli and not *in relation to inputs from afferent neurons* (the latter is considerably harder to do), it is important to keep in mind that there are several stages of neural computation between sensory stimuli and response further down in the sensory processing pathways. This means there is usually only indirect evidence of the actual neural level computations (Kouh & Poggio 2008). Thus it can be hard to know at which stage (or stages) in e. g. sensory processing normalization takes place which poses an additional difficulty when trying to unmask the biophysical mechanisms.

3.3.2 Normalization and neural coding

Normalization is proposed to have several advantages for neural processing where most are related to coding efficiency (Carandini & Heeger 2012). A very important feature of normalization is the dynamic adjustment of the range in which neurons can respond discriminatively on input. Assume e. g. (a bit simplistic) neurons or groups of neurons encoding the values of monetary rewards in their firing rate and that the dynamic range of a neuron is 0-200 Hz. This means without normalization it would be hard for a single neural system to discern both one dollar from two dollars and one million dollars from two million dollars (Carandini & Heeger 2012) without using separate systems for different input magnitudes. However a normalizing circuit will give approximately twice the activity for the more favorable option in both cases. It should though be noted that normalization is not always complete, so some information of the absolute input values might also be left in the output.

Normalization has also been shown to facilitate discrimination by a linear classifier (Luo et al. 2010), which is important for example in odor or object recognition. This can be compared to how normalizing the input can improve performance of an artificial neural network classifier. Schwartz and Simoncelli (2001) have also suggested that normalization reduces statistical dependences in neuronal responses to natural images that cannot be eliminated by linear operations. This would be consistent with the “efficient coding hypothesis” that states that the role of early sensory processing is to remove statistical redundancy in neural signals. Also for odor representations in the fruit fly normalization is shown to result in increased statistical independence (Olsen et al. 2010).

Normalization is also proposed to play a role for decoding probabilistic neural representations; if the outputs are normalized to sum up to one, mean and variance can in a later stage be computed simply as weighted sums of the firing rates (Beck et al. 2011).

3.3.3 Normalization in invertebrates

Evidence of normalization is found not only in more complex nervous systems, but also in invertebrates such as the fruit fly or locust. A study by Olsen et al. (2010) shows the existence of normalization in the fly olfactory system: The response of a second order projection neuron (PN), that receives input from olfactory receptor neurons (ORNs) is shown to be normalized by the activity of other olfactory receptor neurons. For a single neuron driven only by a "private odor" (that does not activate any other PNs) the projector neuron response is accurately described by Equation 3.5.

$$PN = R_{max}\frac{ORN^{1.5}}{\sigma^{1.5} + ORN^{1.5}} \tag{3.5}$$

When a "public odor" that drives other receptor neurons is presented together with the private odor, the response is accurately described by Equation 3.6. Here the normalizing factor s is shown to be linearly proportional to the local field potential (LPF) in the olfactory bulb. Note the similarity to Equation 3.1.

$$PN = R_{max}\frac{ORN^{1.5}}{\sigma^{1.5} + s^{1.5} + ORN^{1.5}} \tag{3.6}$$

In the fruit fly olfactory system the normalizing suppression seems to be the result of presynaptic lateral inhibition between ORNs (Figure 3.3). Papadopoulou et al. (2011) also shows the existence of a normalizing negative feedback loop in the locust mushroom body where the divisive suppression is shown to originate from a non-spiking "giant" interneuron. However, since the neural circuitry is different in primate cortex those mechanisms cannot be directly translated to cortical circuits.

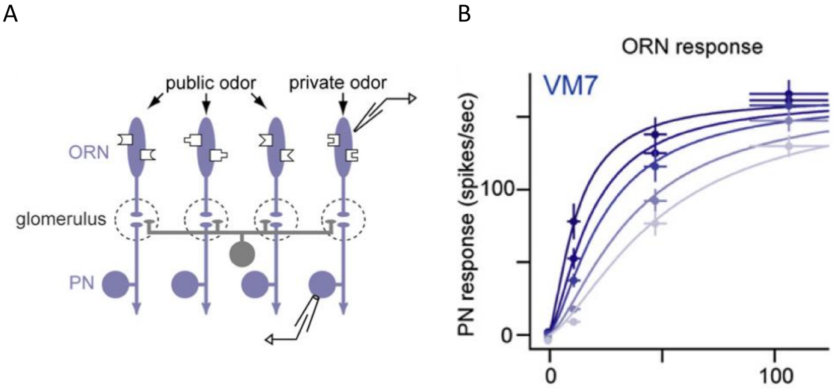

Figure 3.3: Normalization in invertebrates. A) Schematic picture of the neural circuit responsible for normalization in the fly olfactory system. B) IO curves for a projector neuron in presence of increased amounts of a public odor. Fits are to Equation 3.6 with parameters: R_{max} = 163, σ = 12.4 and s = 10.63 LFP. Figures and parameters from Olsen et al. (2010).

3.3.4 Normalization in vertebrate sensory processing

The normalization model was first developed to explain non-linear phenomena for simple neurons in primary visual cortex. Since then normalization have been shown to act at multiple (perhaps all) stages in the visual system (Reynolds & Heeger 2009). The normalization model does e. g. accurately describe how photoreceptors adjust their dynamic range to discount mean light intensity, and can approximate a neural measure of visual contrast (Carandini & Heeger 2012). The response of a photoreceptor obeys the same type of equation found to characterize the response of a projection neuron in the fly olfactory system (Equation 3.7). The exponent n is here found to equal one.

$$R = R_{max} \frac{I}{\sigma + I_m + I} \tag{3.7}$$

Where I is incoming light intensity to the individual photoreceptor and I_m background light intensity. See Figure 3.4 for an illustration of this calculation.

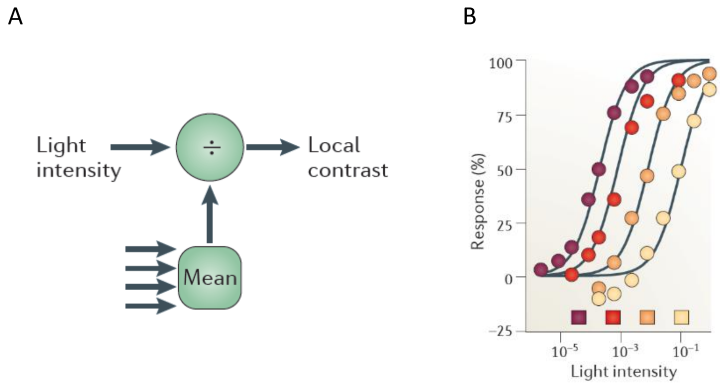

Figure 3.4: Normalization in V1. A) Schematic representation of normalizing circuit. The response from a single neuron is modulated in a divisive way by the summed input to other neurons. B) IO curves for a turtle cone photoreceptor for increased average light intensity. Note the rightward shift on a logarithmic scale; this will adapt the dynamic range to different light conditions. Figures from Carandini and Heeger (2012).

Later signals in retina are thought to feed into a second normalization stage which performs contrast normalization. Here the response is no longer proportional to local contrast, but to the contrast in the area that drives the neuron *relative* the contrast in the region that drives the suppressive field. This is described by Equation 3.8:

$$R_j = \gamma \frac{\sum_k w_{ik} \, C_i}{\sigma + \sqrt{\sum_k \alpha_k C_k^2}} \tag{3.8}$$

Where R_j is neuronal response C_j local contrast, and α_k, and w_{ik} define the summation field and suppression field respectively. This equation e. g. correctly describes surround suppression since it predicts that a smaller stimulus (such as a linear grating), surrounded by a high contrast region, will have its response scaled in a divisive way in proportion to the contrast of the high contrast region.

Reynolds and Heeger (2009) also showed that a slightly extended normalization model for attention can reconcile different findings about how attention modulates neural responses in visual cortex. Normalization is also believed to account for spatial pattern-invariant velocity representation in MT (Carandini & Heeger 2012) and suggested as the mechanism responsible for contrast gain control in the auditory system: Rabinowitz et al. (2011) show that responsiveness of auditory sensory neurons are dynamically adjusted to the contrast of recent stimulation, that is normalization over time. It is also suggested that expanding or compressing the representation of inputs with the help of contrast gain enables a more efficient coding of natural sounds.

3.3.5 Normalization in non-sensory tasks

There is also evidence of normalization beyond the sensory domain. It is for example proposed as a mechanism that can give context-dependent value encoding to guide decision making in primates (Louie et al. 2013). The results suggest that normalized value encoding in decision circuits could play a critical role in decision making since it would give a rescaling of neural activity driven by the value of all choice options, thus implementing a relative value code (compare with the example in Section 3.2.2). It is also shown that the brain displays normalization to the recent history of rewards (which is the reason that you would not be as happy if I give you 100 $ today, if you got 1000 $ yesterday compared to if you got nothing yesterday). This is another example of normalization in a temporal sense.

Normalization is also suggested to be important for flexible multisensory integration, where it could explain cross-modal suppression and provide a unifying computational account of important features of multisensory integration (Ohshiro et al. 2011, van Atteveldt et al. 2014). Moreover it is believed to play a role for causal reasoning, motor control and visual tracking (Beck et al. 2011) by giving near optimal marginalization.

3.4 Proposed mechanistic explanations for normalization

It would indeed be possible to implement normalization with simple neurons that only compute a weighted sum of their inputs. As mentioned previously any continuous function can be approximated arbitrarily well by a multilayer perceptron and an artificial neural net could thus without doubt, be used to approximate a normalizing function as well. However both the structure and learning rule would be far from biologically plausible and we are here looking for mechanisms possible to implement in biological nervous systems. Carandini and Heeger (2012) proposed that normalization could be implemented either by feed-forward or recurrent neural circuitry (Figure 3.5).Their proposed setup includes two different “channels” or mechanisms: One for linear summation of inputs and one that mediate divisive suppression from the normalization pool. A feedback circuit has traditionally also been proposed for primary visual cortex where normalization signals are thought to originate from lateral feedback within V1 (Carandini et al. 2002). Another proposed circuit is a feed-forward circuit with lateral presynaptic inhibition, similar to what appears to be the case in the fruit fly olfactory system.

Kouh and Poggio (2008) also show in a theoretical study that if the mechanisms for divisive gain control and exponential amplification are in place, a canonic neural circuit with different values for the exponents in the normalization model

(Equation 3.4) could implement not only normalization but also winner-take-all, Gaussian and max like operations.

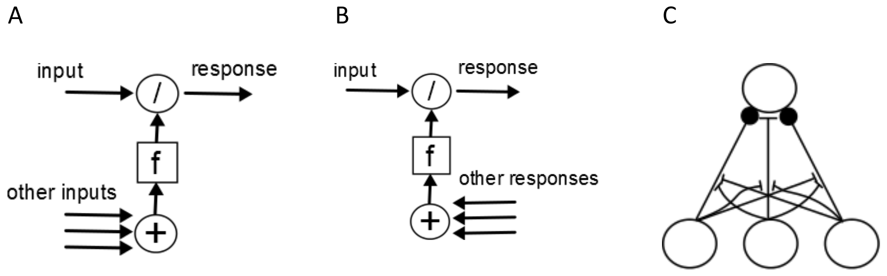

Figure 3.5: Proposed architectures for normalizating circuit. A) Feed forward circuit where other inputs are pooled, and affect the computational unit in a divisive way. B) Recurrent circuit where instead the modulating normalizing signal originate from other responses in the normalization pool. C) Neural circuit with presynaptic lateral inhibition. Note similarity to Figure 3.3 A. Images in A and B drawn with inspiration from Carandini and Heeger (2012), C with inspiration from Kouh and Poggio (2008).

3.4.1 Shunting inhibition

One of the first proposed mechanistic explanations for normalization was “shunting inhibition” (Carandini & Heeger 1994). Normalizing shunting synapses, with an equilibrium potential close to the neurons resting potential, were assumed to be responsible for divisive scaling (and thus the target of modulatory inputs). When activating these synapses the result would be a change in the neuronal membrane conductance, but no or little net ionic current. The cells driving current for subthreshold potentials is indeed scaled divisively by total membrane conductance in accordance with Ohms law ($V = I_{syn}/g$) and this was believed to be transformed into a multiplicative scaling of the input frequency – output frequency relationship.

That changing the total conductance affects the gain of the *subthreshold* membrane potential is indeed true. But it was later showed in theoretical work by Holt and Koch (1997) that when a cells is engaged in sustained firing a change in membrane leak conductance does actually have a subtractive and not divisive effect on output firing rates. This follows since a neuron engaged in sustained firing will have its membrane potential “clamped” above the resting potential by the firing mechanism, which will invalidate the assumption that the shunting conductance does not result in any net current. This subtractive effect of shunting synapses in the absence of noise has been confirmed for biological neurons e. g. by Chance et al. (2002). However the picture changes somewhat in the presence of balanced noise and dendritic saturation (Prescott & De Koninck 2003) which is the reason “shunting inhibition” in an extended sense, in spite of these results is still considered a possible mechanism for normalization.

3.4.2 Shunting 2.0 - Gain modulation from background synaptic input

One of the mechanisms that have both experimental and theoretical support, and is quite widely proposed as a possible mechanistic explanation for the divisive suppression in the normalization model, is gain modulation from balanced background synaptic input (Carandini & Heeger 2012, Chance et al. 2002, Kouh & Poggio 2008, Silver 2010). This mechanism is based on that in addition to the driving input there is also a modulatory "noise channel" (i. e. how large the input fluctuations are) which carry an independent gain modulating signal. Indeed in vivo neurons are constantly bombarded by synaptic input and the functional role of this noisy background activity has long been unknown (Abbott & Chance 2005). It is however well known that background activity affects neuronal response properties by changing the gain and shifting the rheobase of the IO curve. This can be understood intuitively since increasing the amount of balanced noise input will result in no change in average membrane potential but an increased membrane *variance*. This leads to the membrane potential sometimes crossing the firing threshold even for driving inputs which otherwise would result in a membrane potential constantly below threshold (Chance et al. 2002). The mechanism is illustrated in Figure 3.6. Increased membrane variance will also reduce the gain of the IO curve, since this effect is strongest for small inputs and gradually decreases as the driving input grow stronger.

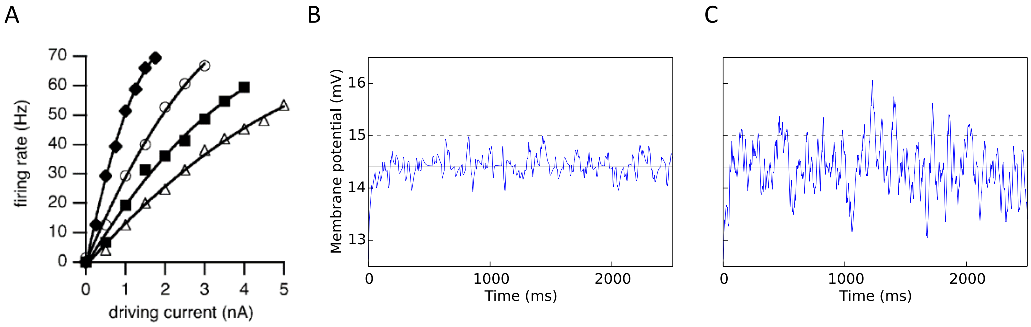

Figure 3.6: Gain change for an in vivo neuron and how increased membrane variance will affect firing for average membrane potentials below rheobase. A) Gain change for an in vivo neuron receiving increased amounts of noisy synaptic input. B) Membrane potential for an IAF neuron injected with a constant current not large enough for it to reach firing threshold as well as balanced noise in form of excitatory and inhibitory Poisson trains. B) During low noise condition the membrane potential does not cross the firing threshold and the neuron remain silent. C) When more noise (here by increasing the frequency of both the excitatory and inhibitory inputs), the membrane potential variance increase; the membrane voltage will now sometimes cross the threshold even if the average membrane potential is the same as in B. Neuron parameters in B and C same as for Model A standard (Table 4.2). Image in A from Chance et al. (2002).

When a neuron is bombarded by balanced synaptic input this will also result in a change in membrane conductance without any net synaptic drive i. e. shunting (Abbott & Chance 2005). This will as discussed in the previous section result in a subtractive shift of the IO curve, for a neuron engaged in sustained firing. The combined effect for a certain parameter range, and if the background activity is indeed carefully balanced is a pure change in neuronal gain (Figure 3.7). The same mechanism is in a theoretical study also shown to work for integrate-and-fire neurons that receive time-variable input (Ly & Doiron 2009).

Critique against this hypothesis is e. g. that it is only valid under quite restrictive conditions (where the balanced and variable background noise is one) which might not be so easy to achieve in the biological nervous system as in carefully controlled experiments. The mentioned studies does indeed not include any biophysically related mechanism for generating the right amounts of noise, but noise is postulated to increase in relation to the activity in a group of neurons, see e. g. Ayaz and Chance (2009). A possible source of noise is though of course the noisy activity of other neurons, and there are evidence that the background activity in cortex is balanced. However, it is not yet known if and how the background activity stays balanced, so as not to introduce an excitatory or inhibitory drive, and which mechanisms that could increase it with the right factor at the right moment to give appropriate divisive scaling.

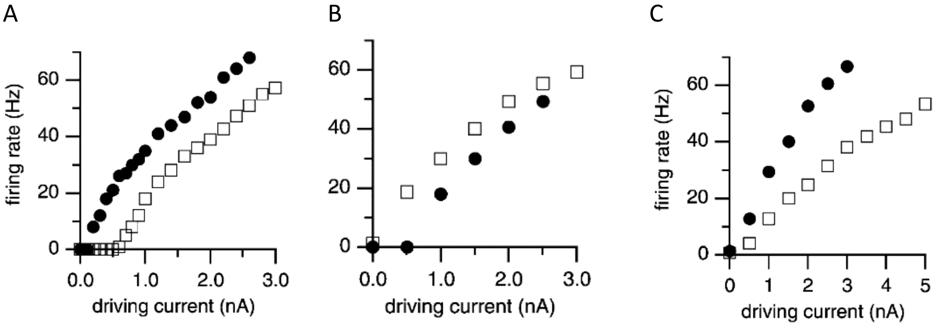

Figure 3.7: Gain change in an in vitro neuron induced by adding increased amounts of balanced noise. A) The effect of adding a constant inhibitory conductance. This effect is the same effect resulting from the shunting introduced by balanced noise. B) The effect of adding increased amounts of balanced noise without the shunting; this gives both an additive shift and a slope change of the IO curve. C) When both those effects are combined the result can be a pure change in neuronal gain. Images from Chance et al. (2002).

3.4.3 Short-term synaptic depression

Another theory is that normalization could be caused by nonlinearities in the afferent inputs to a neuron (Rothman et al. 2009). One such non-linearity is short-term synaptic depression. Carandini and Heeger (2012) argue that if a synapse transmits input to both the studied neuron and neurons in the normalization pool, its effectiveness can be reduced in a way that resembles the divisive effect required for normalization. This was demonstrated in a model by Carandini et al. (2002) where synaptic depression at the thalamocortical synapses was shown to give divisive suppression of neuronal responses. This would in contrast to the prevalent view - where the suppression is assumed to originate laterally from other V1 neurons - imply a feed-forward normalizing circuit. This setup however requires that driving and modulating inputs are mediated by the same synapses and can thus not explain e. g. surround suppression.

Maybe most interesting for the study conducted in this thesis is how Rothman et al. (2009) show that in multi-compartmental models of cortical pyramidal and simple granule cells, short-term synaptic depression converts inhibition-mediated additive shifts of the neuronal response curve, into multiplicative gain change (Figure 3.8). This is the main reason why short-term synaptic depression was included in one of the implemented models versions. It should however be noted that in the study by Rothman et al, the effect of synaptic depression is that not only the gain but also the saturation level of the IO curve is changed; that is the modulatory effect is output gain and not normalization. Another interesting aspect of this is that Murphy and Miller (2003) in a computational study show that by assuming driving input is a non-linear saturation function of stimulus intensity (not specifically because of synaptic saturation) will give rise to very similar effects.

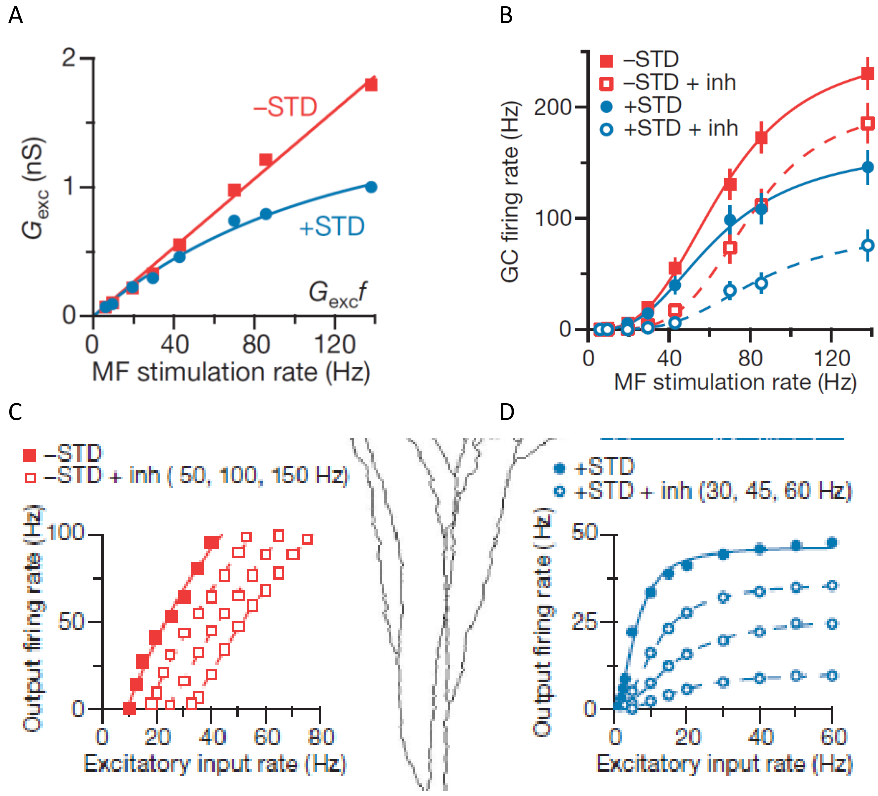

Figure 3.8: Short-term depression as a modulator of neuronal gain. A) Relation between input frequency and excitatory conductance for a simple granule cell. B) Difference between how a modulatory input affects the IO curve with (blue) and without (red) short-term depression (STD). C) Pyramidal cell without STD D) Pyramidal cell with STD. Note the divisive scaling and decreased max output. Figures from Rothman et al. (2009).

3.4.4 Dendritic nonlinearities

In line with the nonlinearity between afferent inputs and neuronal firing rate introduced by synaptic depression, dendritic nonlinearities are also suggested to be able to mediate multiplicative gain changes. When synaptic inputs are located in a distributed fashion they tend to sum linearly owing to the passive properties of the dendrites. But if the synapses instead are spatially clustered it may produce a sufficiently large local depolarization to active nonlinear dendritic mechanisms (Silver 2010). It has been suggested based on theoretical studies that dendritic saturation introducing a non-linear relation between membrane potential and excitatory input will amplify noise mediated gain modulation of sustained rate-coded inputs (Prescott & De Koninck 2003). In a computational study Gabbiani et al (2004) also suggest an implementation of multiplication in a single neuron where two inputs encoded logarithmically are subtracted and the result exponent-tiated through active membrane conductances; that is the neuron performs the computation $a(1/b) = \exp(log(a) - log(b))$. This type of mechanism have also been proposed at network level. The possible role of dendritic nonlinearities will however not be explicitly considered for the models in this thesis since the modelling approach does not include multicompartmental models.

3.4.5 Ongoing activity

Spontaneous activity (the firing of neurons in the absence of any sensory input) has previously been considered as just a byproduct of activity in other parts of the brain with no real purpose (Ringach 2009). It is now believed that the ongoing activity might be a top-down prediction/expectation signal that interacts with incoming input to predict the most likely representation of the world in a probabilistic sense. This would means ongoing activity shapes the response properties of a group of neurons, making them more or less likely to fire for a certain stimulus (Ringach 2009) and could thus be considered a modulatory signal. It has indeed has been shown that the responsiveness of neurons in primary visual cortex is to a large degree controlled by ongoing activity. This could mimic the effect of divisive suppression (Carandini & Heeger 2012). Since only a single hypercolumn is modelled in this thesis, there has not been any intent to try and implement this mechanism.

Concluding remarks on biophysical explanations

In conclusion: There is a considerably larger amount of studies describing normalization functionally or showing theoretically the benefits effective information processing, than there is certainty about the underlying biophysical mechanisms. It is important to note that normalization could be mediated by different biophysical mechanisms for different tasks, species and brain regions so maybe we are not looking for a single mechanism but several. Even in a specific neural circuit and task normalization might be a result of a complex combination of biological biophysical mechanisms and neural circuitry (Priebe and Ferster 2008) so it might not enough to study normalization in the context of driving and modulatory input to a single neuron or computational unit but a larger neural circuit might be needed.

4 Materials and methods

This chapter first describes the inspiration/blue print for the implemented network models in the form of a hypercolumn from a biophysically detailed attractor memory model, and then the structure and details of the implemented model versions. The third part "Evaluation methods" describes the three theoretical models of cortical processing used as a reference during evaluation, as well as the methods used for evaluating the modulatory effects of the model versions.

4.1 Model blueprint

The models implemented in this thesis, are primarily based on the biophysically detailed attractor network model (in the following referred to as "the attractor network") first implemented by Lundqvist et al. (2006). The attractor network is based on the hypercolumnar and minicolumnar organization found in primate cortex (see Section 2.2) and use a group of neurons as the basic computational unit. The network work as an associative memory and perform pattern completion, rivalry and recall while remaining biologically realistic neuron to neuron connectivity ratios and recreating large scale dynamics typically found in cortex (Lundqvist et al. 2011, Lundqvist et al. 2013). The attractor network is constructed from several hypercolumn modules, each composed of a certain number (this vary between different implementation but usually between 9 and 100) of minicolumns. The minicolumns are modeled as groups of recurrently connected excitatory neurons sharing a common input. Each hypercolumn also has a local population of inhibitory cells. The inhibitory population is driven by the excitatory cells in its constituent minicolumns and provide feed-back inhibition to those. The minicolumns are the functional units of the attractor network and depending on if two minicolumns in different hypercolumns are part of the same pattern or not, there are excitatory or inhibitory connections between them. Patterns or "memories" are stored in the network by supervised training that change synaptic weights between minicolumns. A schematic picture of the attractor network and its relation to the more abstract Hebbian attractor memory as well as the hypercolumn model implemented in this thesis is seen in Figure 4.1.

If some minicolumns in the attractor memory model is briefly stimulated by an excitatory input, a pattern (that is a group of connected minicolumns) will activate and stay active for a couple of hundred milliseconds before it spontaneously die out and leave room for other patterns to activate. The temporal dynamics is the result of adaptation and short-term depression. If several minicolumns in a single hypercolumn are stimulated (that is at least two different patterns), there will first be a short transient period with rivalry between patterns, but then one pattern will "win" leaving only one minicolumn in each hypercolumn active (Lundqvist et al. 2006). The feed-back from the inhibitory pool together with the connections from other minicolumns is what cause each hypercolumn to function as a soft WTA module. An "active minicolumn" is also shown to have approximately the same activity level during pattern recall, which together with the WTA dynamics implies that the activity in every hypercolumn is kept constant during an attractor state.

This non-linear behavior constitute the main reason the implemented network model was based on a hypercolumn from the attractor network. If the hypercolumn works as a soft WTA-module for a certain range of inputs, it might be possible to tune/extend such a hypercolumn to also implement normalization. It has indeed been suggested that both these dynamics might be implemented by the same type of circuit implementing some key neuronal computations (Kouh & Poggio 2008). Keeping the average activity constant in a group of computational units is also one of the key features of a normalizing network. This type of structured network also enables using not a single neuron but a group of neurons as the computational unit in a normalization study, which might enable capturing mechanisms not possible to represent in a single neuron. For simplicity and in hope of clarifying the underlying mechanisms of the hypercolumn/soft WTA module, a single hypercolumn is modelled in this thesis.

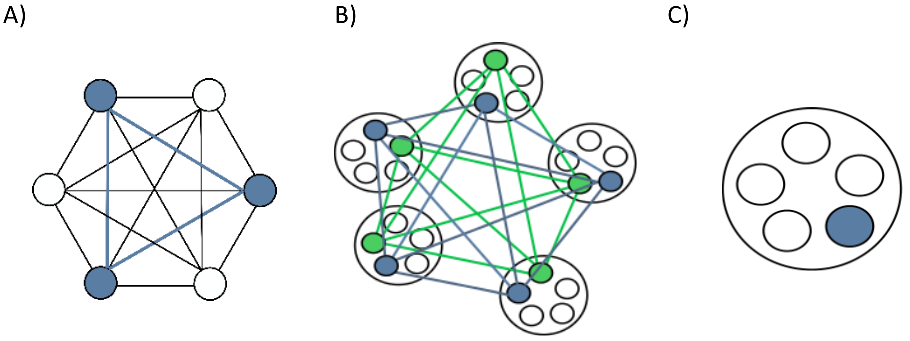

Figure 4.1 A) Hebbian attractor memory constructed with single abstract neurons. The blue neurons constitute a pattern and are connected by stronger positive weights. B) Schematic of the attractor memory model seen from above. A pattern consists of one minicolumn (small circles) from each hypercolumn (large circles) and the excitatory connections are shown for two patterns, blue and green. The basic unit is now a group of biologically realistic spiking neurons, but the logical principle remains the same. C) The network blueprint for the model implemented in this thesis: A single hypercolumn. One active minicolumn is colored to symbolize the WTA functionality. Pictures in B and C drawn with inspiration from Lundqvist et al (2013).

4.2 The network model

The network model implemented and studied in this thesis is based on a hypercolumn/soft WTA module from the biophysically detailed attractor memory network. It is thus a highly structured network modeling layer 2/3 of a single cortical hypercolumn. The hypercolumn is composed of a variable number of minicolumns (1-4 for the experiments reported here). Using a larger number of minicolumns (8-10) was also tried but did not result in any qualitative differences in network processing, so a smaller number was chosen for simplicity. The explicitly modelled minicolumns should be seen as the *active minicolumns* for a certain set of stimuli in a biological hypercolumn constituted of a much larger number of minicolumns. This is a reasonable reduction since most minicolumns in a biological hypercolumn have been found to be silent during a specific set of stimuli.

The network structure is illustrated in more detail in Figure 4.2. Each minicolumn is modeled as a population of $N_E = 30$ excitatory cells with intrinsic recurrent connectivity. The hypercolumn also includes a pool of $N_I = 16$ inhibitory neurons, in common to all the minicolumns. The excitatory cells will as in the original attractor memory model be referred to as “pyramidal cells” and the inhibitory as “basket cells”, which reflect that the excitatory cells in this model are loosely considered to correspond to the cortical excitatory pyramidal cells and the inhibitory cells to fast spiking cortical interneurons. It should however be noted that the cells in the basic versions of this model do not reproduce the specific firing characteristics of these types of cortical cells. The choice was made to keep the number of inhibitory neurons fixed independent of the number of modelled minicolumns. This was done to make the network scale invariant when varying the number of minicolumns, in the sense that each minicolumn will keep the same number of incoming inhibitory connections. This is reasonable because also in a cortical hypercolumn with a smaller number of active (here explicitly modelled) minicolumns, there are others close by that would contribute neurons to the inhibitory neuron pool.

The network structure is a realization of a type of connectivity that lies at the far end of a spectra ranging from networks with no inherent structure (a connection between two neurons is just as likely to occur independently of their relative position) to networks where neurons are clustered and only have connections intrinsically within the cluster. Also more moderate clustering is shown to affect network functionality (Litwin-Kumar & Doiron 2012) and this type of model should be seen as not necessarily representing only the functionality in a cortical hypercolumn; but the minicolumns could in a broader sense correspond to locally connected groups of cells (Lundqvist et al. 2013).

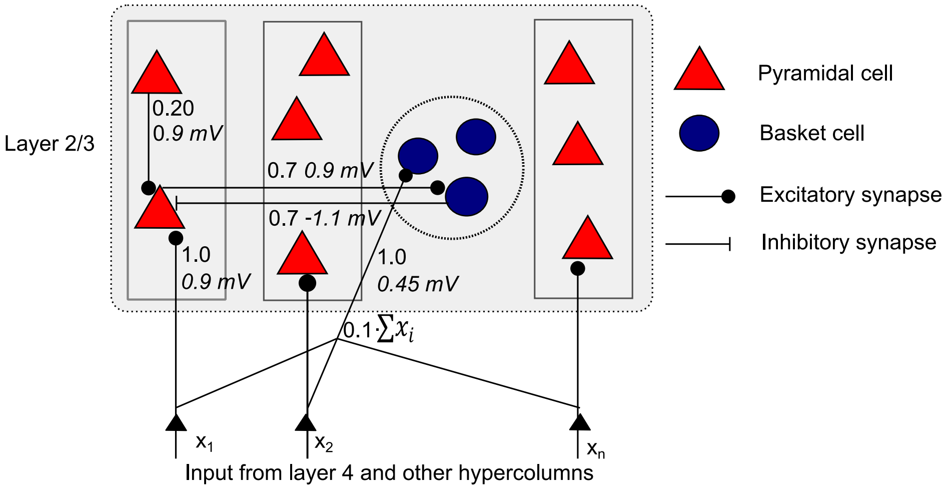

Figure 4.2: Schematic picture of the model structure. The picture depicts a hypercolumn with three minicolumns (rectangles) and the basket cell population (large circle). Input $\bar{x}$ to the network correspond to incoming activity from layer 4 and from layer 2/3 of other hypercolumns. Output is defined as average activity in each of the minicolumns. Numbers next to the lines representing synaptic connectivity is PSP size (cursive) and connection probability. The summation show how input to the basket cells is calculated from driving input to the minicolumns. Picture drawn with inspiration from Lundqvist et al. (2011).

4.2.1 Connectivity

The network is as mentioned highly structured and there are no connections between pyramidal cells in different minicolumns. There are recurrent connections between pyramidal cells in the same minicolumn with a connection probability of 20 %. The network is recurrent also on a larger scale since the inhibitory pool is driven by the activity in the minicolumns, and then provides inhibitory feed-back to those same minicolumns. The probability of a connection between a pyramidal cell in one of the minicolumns and a model neuron in the basket cell population is 70 %, and the same connection probability applies for the inhibitory feed-back. This means that while all minicolumns get different driving input they will all receive the same amount of inhibitory input from the basket cell population. This setup was used also in the attractor network model and should thus be adequate to produce the soft WTA effects observed there.

In addition to the high connectivity found between local excitatory and inhibitory cortical cells, there are also evidence of long range cortical afferent inputs converging on both excitatory and inhibitory cells (Isaacson & Scanziani 2011). To reflect this a feed-forward inhibitory circuit was added as well. This was done by including the possibility to let the basket cell population be driven by a certain percentage, p_{in_bas}, of average driving input to the minicolumns. This means the network, in contrast to the attractor memory model, can implement both a feed forward and a feed-back inhibitory circuit by changing the connection probabilities. This is an advantage since both these circuits are hypothesized to play a role for implementing normalization (compare with Figure 3.5).

In the basic network versions the connection probability was realized by, for each postsynaptic neuron connecting it to p_{pre_post}*N_{pre} randomly chosen neurons from the presynaptic population, where p_{pre_post} is the connection probability between the populations and N_{pre} the number of presynaptic neurons. This means the number of incoming connections are fixed for each neuron, while the number of outgoing connections vary in accordance with a binomial distribution. To study how variability in the network affects its functionality, this was for some model versions changed to instead having a fixed number of outgoing connections and a variable number of incoming connections, which creates a larger variability of the incoming excitation/inhibition for each model neuron.

4.2.2 Input

During simulations each minicolumn receives a driving input of constant Poisson frequency. This represents both input from layer 4 and lateral input from mini-columns in other hypercolumns. The larger portion of this input would be lateral, thus e. g. in the attractor memory model or a cortical hypercolumn correspond to input from minicolumns in other hypercolumns. However in this model there is no functional distinction between these two types of input. The afferent neurons are not modelled explicitly but input to each neuron is approximated as an independent Poisson process of specified frequency. All neurons receive driving input in accordance with a truncated normal distribution of mean I_m and standard deviation 0.1 I_m, constrained to [I_m - 0.2I_m, I_m + 0.2I_m]. As mentioned previously the basket cell population might also receive feed-forward input in addition to the drive from the pyramidal cells. When this is the case a basket cell receive input in accordance with a similar clipped normal distribution.

Noise input

Both pyramidal and basket cells in the network also receive noise input, representing inputs from more distant parts of the nervous system. This is also modelled as independent Poisson processes. The noise input is calibrated to give an excitatory postsynaptic potential (EPSP) that is considerably smaller than the EPSP of the driving input (approximately 0.1 mV) for both pyramidal and basket cells. For pyramidal cells the frequency of the noise input is chosen to bring the pyramidal cells to weak spontaneous firing (1-5 Hz). There is evidence of basket cells receiving less noise so for most model versions the basket cells receive noise that do not bring them to rheobase or no noise at all. The amount of noise to the basket cells was however used to tune some of the network model versions.

4.2.3 Neuron and synapse model

The aim was to use a neuron model as simple as possible, while still able to capture the neuronal features hypothesized to be important for normalization. An IAF neuron model is the least complex model that can implement all the proposed gain change mechanisms discussed in Section 3.4 which do not require a multicompartmental model. Accordingly a leaky integrate-and-fire (IAF) neuron model with conductance based synapses was used for all model versions, except one where this model was replaced with an adaptive IAF neuron model. The subthreshold dynamics of the model neurons are thus described by Equation 2.6 and the synaptic input by Equation 2.10 and Equation 2.12 (repeated in Table 4.2 for convenience).

The neuron parameters are loosely based on the Hodgkin-Huxley neurons of Lundquist et al 2006 translated to an IAF neuron model. This was done by simply using the values of e. g. membrane conductance and the area of the each neuron type to calculate the corresponding value for a point neuron model. It should be noted that this does not render model neurons to reproduce those qualitative differences in the firing patterns between the basket and pyramidal cells seen for the original multicompartmental models. This large reduction in complexity is motivated by that the WTA effects of the attractor memory model have been reproduced without these qualitative differences in firing patterns (Bruederle et al. 2011). There is also neither adaptation nor short-term depression included; the effect of those features were instead tested separately. Another difference is that the same value of the neuronal time constant, 13.5 ms is chosen for both excitatory and inhibitory cells. The refractory period is a bit shorter for the basket cells compared to the pyramidal cell (2 ms in contrast to 3.5 ms) which reflect that basket cells tend to be faster spiking than pyramidal cells. The membrane time constant for both excitatory and inhibitory synapses is 6 ms. It should be noted that the specific neuronal parameters are not hypothesized to have a significant effect on network functionality - mainly because of the low complexity of the neuron model - as long as synaptic weights are tuned to reproduce the same PSP sizes.

Calibration of synaptic weights

The synaptic weights in the network are specified implicitly as the weights needed to produce postsynaptic potentials (PSPs) of specified sizes for the different synaptic connections. The PSP values for the standard model version are equal or close to those used in the attractor network. Tuning of synaptic weights was done by considering the PSP size resulting from a single synaptic event for a model neuron at resting potential. For all models the EPSP from driving input to the pyramidal cells is fixed at 0.9 mV and the EPSP from noise input to pyramidal and basket cells at 0.1 mV. Standard PSP values for feed-forward input to basket cells; excitatory connections between pyramidal and basket cells; and inhibitory feed-back from basket cells to pyramidal cells; are 0.45 mV, 0.45 mV and -1.1 mV respectively. However the latter three were varied in some model versions to study what effect this has on network functionality. The PSP values as well as the synaptic weights which produce those are summarized in Table 4.2. An illustration of the calibration and PSPs of the model neurons are seen in Figure 4.3. Since it is in fact the synaptic weights that are changed during parameter tuning of model versions, the parameters representing those are usually the ones given when describing differences between network model versions in the Results chapter. The different synaptic weights are referenced to as EPSC$_{\text{pre\_post}}$ PSPs are referred to as EPSP$_{\text{pre\_post}}$. Here “pre” and “post” are the pre- and postsynaptic populations. Pyramidal cell population (minicolumn) is abbreviated as pyr, basket cell population as bas, and external population as ext.

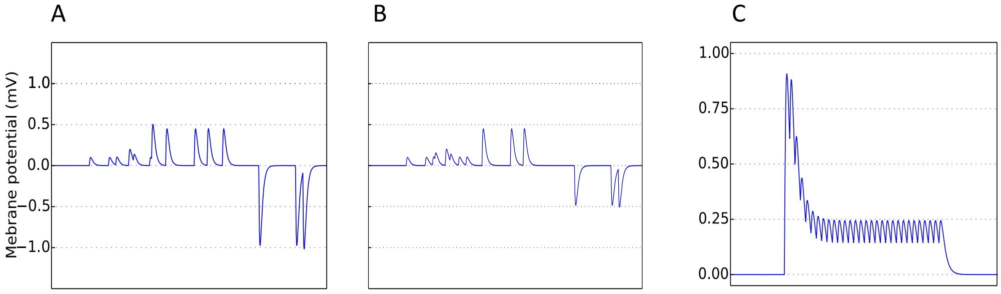

Figure 4.3: PSP calibration and depressing synapses for Model B. A) For a pyramidal cell the PSP from driving input is 0.9 mV, and from noise input 0.1 mV. The IPSP from the inhibitory pool is 1.1 mV. B) For basket cells the EPSP from pyramidal cells as well as external input is 0.45 mV, noise input 0.1 mV and eventual negative recurrent IPSP from other basket cells 0.45 mV. All PSPs are recorded from resting potential. C) The effect of synaptic depression on pyramidal PSPs. Input rate: 50 Hz. Synaptic depression parameters and equations are seen in Table 4.2.

Depressing synapses

In one model version (termed “Model B”) short-term synaptic depression is included for the driving input to the pyramidal cells. Including synaptic depression on all pyramidal-pyramidal synapses - that is also for the recurrent connections - was tested separately for this model version. There is no depression for feed forward input to basket cells or for synapses between excitatory or inhibitory cells (and vice versa) since there is no evidence that synaptic depression is present in cortex for these connections. Depressing synapses are modelled according to Equation 2.14 (repeated in Table 4.2) with a time constant for recovery of 200 ms.

For Model B a population of 200 excitatory cells are created to represent the neurons from which the pyramidal cells in layer 2/3 get their input. These neurons are modelled as “parrot neurons” that transform every incoming spike to an outgoing spike without modelling any intrinsic neuronal dynamics. This population which receive Poisson input will thus as well spike as a Poisson source. The parrot neurons were necessary because it is not possible in NEST to create a depressing synapse between a Poisson generator node and its postsynaptic nodes. The connectivity between the layer of “input neurons” and the pyramidal cells in the minicolumn is 50 %. This means every pyramidal cell will receive input from approximately 100 neurons. When the input to the network is specified as X Hz this is realized by letting each neuron in the input layer spike with Poisson frequency X/N$_E$ Hz, to render a total average input to the minicolumn (X/N$_E$)*N$_E$ = X Hz. This definition was done for simple comparison with the other model versions. The reason a quite large population of post-synaptic neurons need to be modelled explicitly, is that a synapse with short-term depression have a limit to how much current it can transport over the cell membrane; replacing this group of neuron with fewer neurons spiking faster, would result in a very limited input to the network.

4.2.4 Biological realism

The aim of this thesis is to gain deeper understanding into how this type of structured recurrent network behaves functionally and whether the mechanisms it implements may be related to how normalization is achieved in biological nervous systems. Biological realism is achieved by using biological data as a constraint during the network design (e. g. neuron parameters, PSP sizes, connectivity and structure present in cortex). But within these “biological bounds” the network parameters are varied to see how this affect network functionality. It should however be noted that the aim of this quite abstract model is not to try and reproduce realistic cortical firing characteristics, such as interspike intervals and oscillatory activity, but focus lies on how the population firing rate of a group of recurrently connected cells is modulated by the firing rates in other cell assemblies.

To find out if and when increased complexity do change model functionality there were several tests conducted where more complex features were added to the basic model version, to see if and how this might change the properties of the network. To start simple and gradually increase complexity was hoped to help elucidate which features that are important for normalization/gain-control in the model.

Biologically plausible parameter space

The connection probabilities and the PSPs for the standard model versions are the same or similar to the original attractor memory model, where biological constraints on these values are mainly from Thomson et al (2002). Since we here wanted to vary parameters to see how this affects network functionality, it was necessary to consider in which range it is meaningful to do so. Since PSP and connection data is quite uncertain (there are often only a few cells investigated from one or two species) the choice was made to use these data as a guidance rather than an exact postulation. This means a quite broad range of possible PSP values was considered: in the range from 0.075 mV to 7.5 mV for all connection types. This is a range that contains all the values found by Thomson et al (2002), and a deviation from the average both up and down in an order of magnitude of 10. This should though only be seen as a broad constraint which enable ruling out truly unrealistic values and not that all values within this range are equally plausible. But it was for the for the purpose of finding tendencies and patterns for how changing different parameters affects functionality deemed better to search a too large area than a too small. It is always possible to later use only values within a subpart of this area.

4.2.5 Simulator

There are several simulators available to construct and simulate neural networks, which means it is not necessary to implement neuron and synapse models from scratch. The choice of simulator for implementing and running the models designed and evaluated in this thesis was NEST. NEST is a thoroughly tested and efficient simulator for large networks of point neurons or neurons with a few compartments (Gewaltig et al. 2012). As a high level interface to the NEST simulator PyNEST (Eppler et al. 2008) was used which gives simple integration with python code. NEST includes the neuron and synapse models used in the network model so no new models were constructed. Since comparisons between simulators have not shown qualitatively different results when implementing the same networks on different simulators (and only slight quantitative differences) (Brette et al. 2007) the choice of simulator is not believed to influence the results in any significant way.

Randomness

For generating random connectivity and Poisson trains the built in random number sources of NEST were used. Membrane conductance, initial membrane potential and input was randomized by the means of the numpy.random module. The membrane conductance for each neuron is sampled from a truncated normal distribution with mean C_m and standard deviation of 0.1 C_m constrained to the range [C_m - 0.1C_m, C_m + 0.1C_m]. The driving input to each neuron is sampled from a similar truncated normal distribution with mean I_m and standard deviation 0.1 I_m, constrained to [I_m - 0.2I_m, I_m + 0.2I_m]. The membrane potential at setup ($t = 0$) is also randomized but has a broader distribution with mean 5 mV and standard deviation 5 mV constrained to [0, 10] mV.

Because of variability in neuron parameters and the randomized connections there will be slight quantitative differences if the same tests are run for different network setups (i. e. different random seeds). The choice was made to not make one setup and keep this for all network tests, but running the tests for a range of setups with different random seeds. This was done to ascertain that not only one network version that might show functionality different to most realizations of the random setup was studied. In practice a smaller subset of random seeds were used for producing the results presented in the report, but the network structure have been tested on a larger set without showing qualitative different results.

4.2.6 Differences from the attractor network

The most significant differences between the model implemented here and the attractor network is that only a single hypercolumn is modelled. This will mean the larger network dynamics as well as temporal dynamics of different attractors activating after each other are lost. On the other hand we gain in simplicity. The attractor memory properties are not important for normalization, but it should be noted that the excitatory and inhibitory connections between minicolumns in different hypercolumn do play a part in the WTA behavior as well. Using IAF neurons and that short-term depression and adaptation is not included on model neurons in the basic model versions, are also significant differences. Considering the input, the model implemented in this thesis receives a stationary input and wait until the network reach a steady state before reading the output. This in contrast to the attractor network where a brief input stimulates the network, which then keeps active without any external input. The stationary input could for a hypercolumn that is part of an attractor network be seen as corresponding to input from other minicolumns in the same or rivaling patterns during an attractor state. However, the input to the hypercolumn implemented here could also for a different setup such as a network used for probabilistic calculations, correspond to input representing confidence in different object features.

Model versions

Four different main model versions were studied, all based on Model A but with some changes in connectivity or synapse dynamics:

- Model A: Basic model version with both feed-forward and feed-back connectivity
- Model A1: Feed-forward inhibitory circuit only.
- Model A2: Feed-back inhibitory circuit only.
- Model B: Short-term synaptic depression included on driving synapses.

Model A1 and A2 were studied to gain insight into how the different inhibitory circuits might contribute to the modulatory effects of the network. These insights were then used to tune Model A before adding additional features such as increased variability in the network; recurrent connectivity in minicolumns and basket cell populations; as well as model neurons with more realistic firing characteristics. Model and neuron parameters are summarized in Table 4.2. Model versions are summarized Table 4.3.

Table 4.2: Summary of network setup and model parameters.

Neuron and synapse model	
Neuron model	Integrate-and-fire neuron. NEST model: “iaf_cond_exp”. See also section 2.4.4.
Subthreshold dynamics:	$\tau_m \frac{dV}{dt} = -V - R_m I_{syn}(t)$
Neuron model adaptive neurons	Adaptive integrate and fire neuron modelled according to Brette and Gerstner (2005). NEST model: “aeif_cond_exp”. See also section 2.4.4.
Subthreshold dynamics	$C_m \frac{dV}{dt} = -g_L(V - E_L) + g_L \Delta_T \exp\left(\frac{V - V_\theta}{\Delta_T}\right) - \omega + I$ $\tau_w \frac{dV}{dt} = a(V - E_L) - w$ $when\ spike\colon\ w \rightarrow w + b$
Synapse model	Exponential shaped conductance based synapses, with infinite rise time. See also Section 2.4.5. $I_{syn}(t) = \sum_j w_{ij} \cdot g_{syn}\big(V(t) - E_{syn}\big)$ $g_{syn}(t) = \begin{Bmatrix} \bar{g}_{syn} exp\left(-\frac{(t - t_s)}{\tau_{syn}}\right) & for\ t \geq t_s \\ 0 & for\ t < t_s \end{Bmatrix}$
Synaptic depression model	Synaptic depression is modelled according to the phenomenological model described in Tsodyks et al. (1998), NEST model “tsodyks2_synapse”. See also Section 2.4.5. $\frac{dx}{dt} = -\frac{z}{\tau_{rec}} - u\,\delta\big(t - t_{sp}\big)$ $\frac{dy}{dt} = -\frac{y}{\tau_{in}} + u\delta\big(t - t_{sp}\big)$ $\frac{dz}{dt} = \frac{y}{\tau_{in}} - \frac{z}{\tau_{rec}}$ Where x represent the available, y the active and z the inactive part of the neurotransmitter.

Populations

Name		Size
E_i	Excitatory populations	N_E = 30 Number of excitatory populations, n_{mc}, = 1-8
I	Inhibitory population	N_I = 16
N	External noise (long range)	N_E* N_{mc} + N_I Modeled as independent Poisson processes.
D	Driving input	N_E* N_{mc} Modeled as an independent Poisson processes for each neuron receiving input.

Connections

Source	*Target*	*Connection probability*	*PSP*	*Weight (peak conductance)*
E_i	E_i	0.2	0.9 mV	0.17 nS
E_i	E_j	0.0	0.9 mV	0.17 nS
E_i	I	0.7	0.45 mV *(0.075 - 7.5)*	0.009 nS *(0.014 - 0.16)*
I	E_i	0.7	- 1.1 mV *(0.075 - 7.5)*	- 2.6 nS *(0.17 - 40)*
I	I	0.0	- 0.45 mV	- 0.02 nS
N	E_i	1.0	0.1 mV	0.02 nS
N	I	1.0	0.1 mV	0.0022 nS
D	E_i	1.0 *	0.9 mV	0.17 nS
D	I	1.0 *	0.45 mV *(0.075 - 7.5)*	0.009 nS *(0.014 - 0.16)*

*All neurons get input but *input frequency* is normally distributed around mean input, to reflect that the number of incoming connections vary for biological neurons. In parenthesis: range of values considered during parameter searches.

Additional network parameters

Abbreviation	Description	Unit	Value
$noise_{pyr}$	Poisson frequency for noise input to pyramidal cells	Hz	0
$noise_{bas}$	Poisson frequency for noise input to basket cells	Hz	5200
$C_{m\_rsd\_bas}$	Relative standard deviation membrane conductance pyramidal cell	% of mean	10
$C_{m\_rsd\_pyr}$	Relative standard deviation for membrane conductance pyramidal cell	% of mean	10

$\mathrm{Input}_{\mathrm{rsd\_pyr}}$	Relative standard deviation membrane conductance pyramidal cell	% of mean	10
$\mathrm{Input}_{\mathrm{rsd\_bas}}$	Relative standard deviation membrane conductance pyramidal cell	% of mean	10
N_{E_4}	Number of layer 4 parrot neurons in Model B	-	200

IAF model neuron parameters

Abbreviation	Description	Unit	Excitatory cell “Pyramidal cell”	Inhibitory cell “Basket cell”
E_L	Resting membrane potential	mV	0.0	0.0
E_{ex}	Excitatory reversal potential	mV	120.0	120.0
E_{in}	Inhibitory reversal potential	mV	- 10.0	- 10.0
C_m	Membrane capacity	pF	70.0	7.5
τ_m	Membrane time constant	ms	13.5	13.5
τ_{ref}	Refractory period	ms	3.5	2.0
V_{th}	Spike threshold	mV	15.0	15.0
V_{reset}	Reset membrane potential	mV	0.0	0
τ_{syn_ex}	Time constant excitatory	ms	6.0	6.0
τ_{syn_in}	Time constant inhibitory synapses	ms	6.0	6.0

Adaptive IAF model neuron parameters

Abbreviation	Description	Unit	Excitatory cell “Pyramidal cell”	Inhibitory cell “Basket cell”
E_L	Resting membrane potential	mV	- 61.71	- 56.0
E_{ex}	Excitatory reversal potential	mV	0.0	0.0
E_{in}	Inhibitory reversal potential	mV	- 80.0	- 85.0
C_m	Membrane capacitance	pF	179.0	6.88
τ_m	Membrane time constant	ms	16.89	15.64
τ_{ref}	Refractory period	ms	3.5	2.0
V_{th}	Spike threshold	mV	- 54.36	- 52.7

V_{reset}	Reset membrane potential	mV	- 60.7	- 72.5
τ_{syn_ex}	Time constant exc. synaptic exponential function	ms	17.5	6.0
τ_{syn_in}	Time constant inhibitory synaptic exponential function	ms	6.0	-
ΔT	Slope factor	mV	0.0	0.0
τ_w	Adaptation time constant	ms	196.0	0.0
a	Subthreshold adaptation	nS	0.0	0.0
b	Spike-triggered Adaptation	pA	0.0132	0.0

Synaptic depression parameters

Name	Description	Unit	Value
u	Release probability	-	0.5
τ_{rec}	Recovery constant synaptic depression	ms	200
τ_{fac}	Recovery constant facilitation	ms	0
	Weight adjustment: To give the same PSP for a single synaptic event when synaptic depression is included, all weights are multiplied by two in synapse models including synaptic depression.		

Table 4.3: Summary of model versions. Parameters or features not mentioned are the same as for Model A standard.

Name	Description
Model A standard	Model A with standard parameters from Table 4.2
Model A1	Connectivity between all minicolumns and basket cell population (p_{pyr_bas}) is set to zero.
Model A2	Feed-forward input to basket cell population (p_{in_bas}) is set to zero.
Model A tuned	Larger IPSP size. IPSC$_{bas\_pyr}$ = 12.0 pS, EPSC$_{ext\_bas}$ = 0.007 pS, EPSC$_{pyr\_bas}$ = 0.005 pS, p_{in_pyr} = 0.05, noise$_{bas}$ = 1250 Hz.
Model A adaptive tuned	Tuned similarly to Model A tuned. IPSC$_{bas\_pyr}$ = 15.0 pS, noise$_{pyr}$ = 2000.0 Hz, noise$_{bas}$ = 10 000 Hz, EPSP$_{pyr\_bas}$ = 0.005 pS.
Model B	Short-term depression added. IPSC$_{bas\_pyr}$ = 2.6 pS, EPSC$_{ext\_bas}$ = 0.006 pS, EPSC$_{pyr\_bas}$ = 0.005 pS, p_{in_bas} = 0.05. $C_{m\_pyr\_rsd}$ = 0.25, $C_{m\_pyr\_rsd}$ = 0.25. Includes variation in number of incoming connections.
Model B tuned	Short-term depression added. IPSC$_{bas\_pyr}$ = 12.0 pS, EPSC$_{ext\_bas}$ = 0.005 pS, EPSC$_{pyr\_bas}$ = 0.004 pS, p_{in_pyr} = 0.02. $C_{m\_pyr\_rsd}$ = 0.25, $C_{m\_pyr\_rsd}$ = 0.25. Includes variation in number of incoming connections.

4.3 Evaluation methods

This section describes the three theoretical models of cortical processing used as a reference during the network evaluation (in the following referred to as "the theoretical models") and the tests used to evaluate the network model versions normalizing ability. For reference these tests are applied to the theoretical models, and the results confirm that a normalizing network has several important information processing features not present for the other theoretical models.

Network input and output

An input to the network $\bar{x}$ = [x_1, x_2, .. x_n] is defined as the driving input to its minicolumns where x_i is input to minicolumn i. The output $\bar{y}$ = [y_1, y_2, .. y_n] associated with a specific input is defined as the average activity in the network's minicolumns after steady state activity is reached, where y_i is activity in minicolumn i. The output is evaluated by simulating network activity for t_{sim} = 500 ms, and then averaging the activity in each minicolumn over the [$t_{st,}$ t_{sim}] (Equation 4.1), where t_{st} is the time where the network is assumed to have reached a steady state activity. For all implemented model versions t_{st} = 50 ms.

$$y_j \equiv \frac{\sum_k n_{jk}^s}{t_{sim} - t_{st}} \tag{4.1}$$

Here the sum goes over all neurons in minicolumn j and n_{jk}^s is the number of spikes emitted by neuron k. The average activity in the inhibitory pool is calculated in the same way. The average activity for the hypercolumn, H_{av}, is defined as the average activity over all its constituent minicolumns (Equation 4.2). Note that the activity in the inhibitory pool is not included here.

$$H_{av} \equiv \frac{\sum_j y_j}{n_{mc}} \tag{4.2}$$

Here n_{mc} is the number of minicolumns in the network.

4.3.1 Evaluating normalization

To evaluate the network models, three different phenomenological models of cortical processing observed in biological nervous systems were used as a reference: a) normalization b) output gain control and c) subtractive inhibition (see Theory chapter for references). The definitions of those models in the context of the implemented network model and a summary of their respective modulatory effects, are seen in Table 4.4. The driving input to a minicolumn/-computational unit is defined as the direct input to that minicolumn, and the modulatory input as the average of the driving inputs to other minicolumns/-computational units. For normalization this type of setup follows straight from the definition of the normalization model (see Section 3.3.1). For the other theoretical models their definition only incorporate how increased modulatory input affect the output not where it comes from. However since for the network studied in this thesis, this will indeed be from the input to the other minicolumns, the same origin of the modulatory input is assumed for all theoretical models.

For a specific input to the network, there are two important dimensions that might affect system performance: First there might be a difference in the functional dependence of outputs on inputs, depending on if the individual minicolumns get input similar in magnitude or if some minicolumn get a much larger input - that is *input relations*. Second, an input vector with fixed input relations might be processed differently depending on *input magnitude*. To capture these two dimensions an input vector to the network can be described as:

$$\bar{x} = a[c_1,\ c_2,\ ..c_n],\ \sum_i c_i = 1 \tag{4.3}$$

Where $\bar{c}$ represent input relations and a input magnitude. A perfectly normalizing system will of course give a normalized output for all possible values of a and $\bar{c}$, but since a neural system might be normalizing for only a subset of input relations and input magnitudes the evaluation is designed with those two aspects in mind.

Two different tests were used to evaluate the network model versions: The “IO curve test” (IO test) and the “Fixed Input Relations test” (FIR test). Both these tests were used to estimate from different perspectives, how well the function from inputs to outputs implemented by the network correspond to the normalization model.

The IO test

To evaluate the networks normalizing ability of over a range of input relations and magnitudes the IO test study how the IO curve of one “studied minicolumn” is changed for different amounts of modulatory input. This type of test is standard in previous normalization and gain control studies (see e. g. (Carandini & Heeger 2012, Chance et al. 2002, Olsen et al. 2010, Rothman et al. 2009, Silver 2010). In the context of this thesis the IO test was performed by plotting a family of IO curves for one studied minicolumn for different amounts of input to the other minicolumns (that is for increased inhibitory feed-back from the basket cell population). Each curve is produced by evaluating the output from the studied minicolumn for a range of inputs, while keeping the input to the other minicolumns at a constant value. The “input to the other minicolumns”, I_{other} is for a network with more than two minicolumns defined as the averaged input to the other minicolumns. The input to the individual minicolumns is not equal but distributed equidistantly in the range [$0.2I_{other}$, $0.8I_{other}$]. This since it is quite unlikely that all other minicolumns would receive the exact same amount of input in a cortical hypercolumn.

The result of the IO test was evaluated by comparing the effects on the IO curve for increased modulatory input to the theoretical models. Focus was especially put on eventual gain change; change in max output (R_{max}); the input which give half max(σ); and subtractive offset (β). To quantify this a sigmoidal Hill type function (Equation 4.4), can be fitted in the least squares sense to each set of data points, see e. g. (Olsen et al. 2010, Rothman et al. 2009, Silver 2010).

$$f(I_i) = R_{max}\frac{(I_i - \beta)^n}{\sigma^n + \ (I_i - \beta)^n} \tag{4.4}$$

For a normalizing network there will be a clear increase in σ (since the additive term in Equation 4.3 here will turn out as σ), while β and R_{max} remain approximately constant. For output gain on the other hand σ and β will stay approximately constant, but R_{max} will be scaled down for increased modulatory input. If on the other hand R_{max} and σ remain constant while there is primarily an increase in β for increased input to the other minicolumns, it can be concluded that the modulatory effect of the network is primarily subtractive.

Since normalization and output gain models will result in changed gain of the response curve for increased modulatory input, this was also evaluated. The gain of the IO curve was calculated as the average derivative of the fitted function between output values y_{start} and y_{stop}, see e. g. Rothman et al. (2009), where y_{start} and y_{stop} can be varied to evaluate the gain for different parts of the IO curve. Normalized gain for each IO curve was then calculated as the gain for the specific curve, divided by the gain of the curve corresponding to the smallest modulatory input.

In the IO test we only study the output from one minicolumn. In addition this test does not allow varying input relations and input magnitude freely but a certain input magnitude is tied to a specific relation between the inputs. It was as well deemed important to study how an input vector with *fixed input relations* is processed for increased input magnitude. This brings us to the second test used for the network evaluation.

The Fixed Input Relations test

To study how the networks process a specific input vector - that is all outputs for a specific input - a second test where the input relations are held constant, while the input magnitude is varied was constructed for the purpose of this project. This test will be referred to as the "fixed input relations test" (FIR). The result is visualized as a graph (denoted FIR graph, see also Figure 5.1) were the outputs from the network are plotted as a sequence of histograms for increased input magnitude. The histograms from the FIR test can show for which input magnitude normalization starts up and breaks down, and also help clarify how input relations and magnitude affect the functionality of the network separately

This test thus enables evaluating in which range of input relations and input magnitudes a network approximate normalization, even if it is not perfectly normalizing. This is something not possible with the more standard IO test. For the purpose of a quantitative comparison between different model versions this test is also quantified for some specific model vectors (see Section 4.3.3).

4.3.2 Normalization vs output gain-control and subtractive inhibition

This section study the theoretical models in the light of the IO test and the FIR test. The results of performing these tests for the theoretical models are seen in Figure 4.4.

IO test theoretical models

Considering the IO test, important to note is that both the normalization and output gain model will give reduced gain (i. e. sensitivity to an increase driving input) of the IO curve, for an increased modulatory input. A difference is though, that for normalization it is still possible to reach the previous max output value (even if this will require a larger driving input) while for output gain the max output is changed together with the gain. This follows from the definition of output-gain where the R_{max} is scaled by a constant for a specific modulatory input. Another qualitative difference between normalization and output gain is that for normalization the point on the horizontal axis where half max activity is reached, is shifted to the right with increased modulatory input, while for output gain it stays the same.

Table 4.4: Definitions of the theoretical models of cortical processing, and their modulatory effects.

Normalization

$$f_i(\bar{x}) = R_{max} \frac{x_i^n}{\sigma^n + g([x_1, x_2, \dots x_m]) + x_i^n} \tag{4.5}$$

$$g(\bar{x}) = d \sum_{j \neq i} x_j^k$$

I Gain change of the IO curve for increased modulatory input.

II No change in R_{max}, but the driving input needed to reach saturation is increased for increased modulatory input.

III Average activity in hypercolumn approach a constant value for increasing input magnitudes.

IV For fixed relationships between the inputs, the relationships between the outputs show no dependence on input magnitude.

Output gain

$$f_i(\bar{x}) = R_{max} \cdot g([x_1, x_2, \dots x_m]) \cdot \frac{x_i^n}{\sigma^n + x_i^n} \tag{4.6}$$

$$g(\bar{x}) = \frac{1}{(1 + d \sum_j x_j)}$$

I Gain change for lower output rates for increased modulatory input (gain is scaled multiplicatively, since the whole function is multiplied by a constant, $f(x) \rightarrow af(x)\ gives\ f'(x) \rightarrow af'(x)$

II R_{max} is decreased for increased modulatory input.

III The point on the horizontal axis where half max is reached remain the same for increased modulatory input.

Above features follow from that the IO curve is scaled vertically by a constant for a specific modulatory input.

IV Relations between outputs do not stay the same for increased input magnitude. For sufficiently input magnitudes the outputs will all saturate and thus the distinction between them is lost.

V Average output vary with increased input magnitude.

Subtractive inhibition

$$f_i(\bar{x}) = R_{max} \frac{(x_i - g([x_1, x_2, \dots x_m]))^n}{\sigma^n + (x_i - g([x_1, x_2, \dots x_m]))^n} \tag{4.7}$$

$$g(\bar{x}) = d \sum_j x_j^m$$

I No gain change for increased modulatory input when considering points of similar output value.

II No change in R_{max} for increased modulatory input.

III Curve keep its shape but is shifted to the right on the horizontal axis for increased modulatory input.

Since subtractive inhibition is defined as subtracting a constant amount from the independent variable before calculating the output, above features follow directly from the definition.

IV Relations between outputs do not stay the same with increased input magnitude. For sufficiently large input magnitudes all non-silenced inputs will saturate and the distinction between them is lost.

V Average output vary with increased input magnitude.

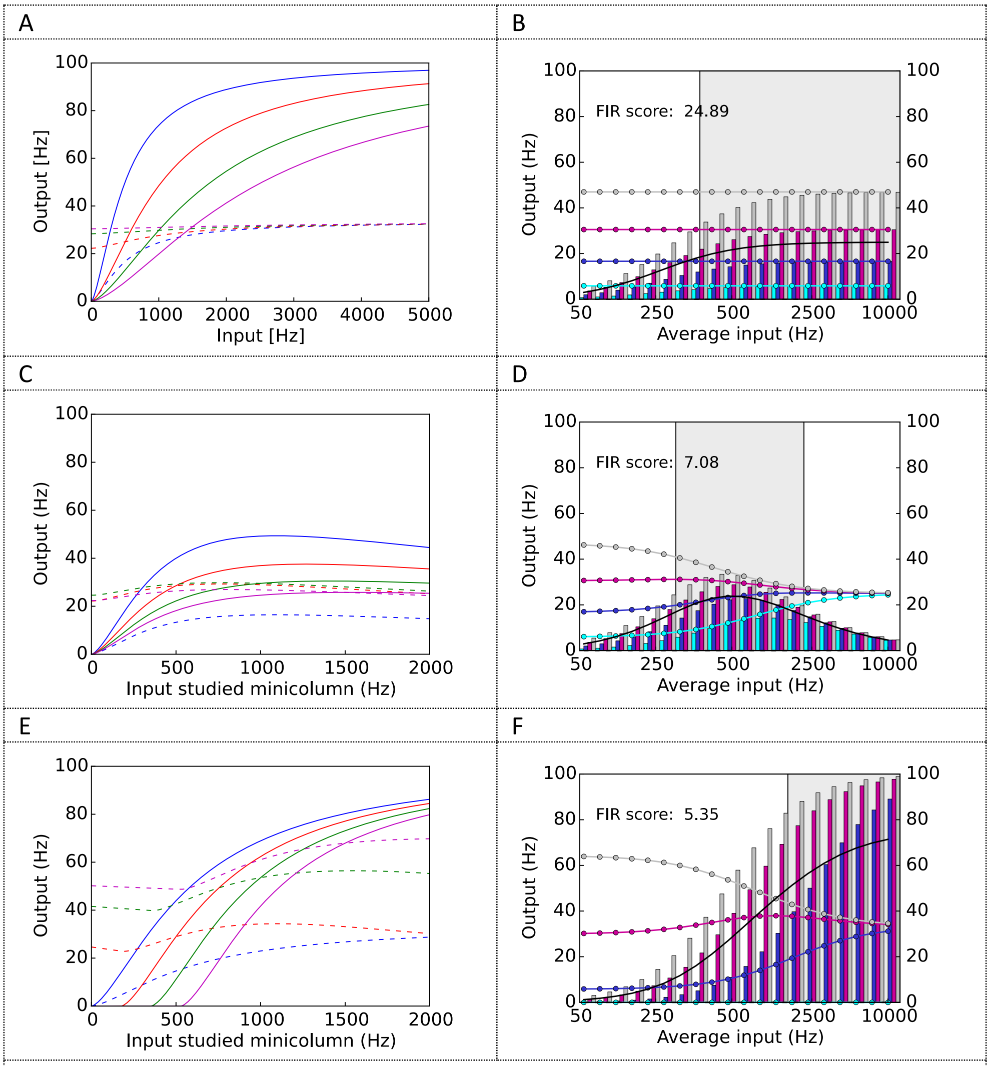

Figure 4.4: Illustration of the modulatory effects of the theoretical models. All plots are produced by the equations found in Table 4.4 with R_{max} = 100, σ = 500 and n = 1.5. Continuous lines: activity with zero modulatory input. Dashed lines: average activity for all computational unit. Modulatory inputs: 500, 1000 and 1500 to two additional computational units respectively. A, C, E) IO test. B, D, F) FIR test $\overline{x}$ = a [0.1, 0.2, 0.3, 0.4] for increasing a. A, B) Normalization (Equation 4.5) with d = 1. C, D) Output-gain control (Equation 4.6) with d = 0.0005 and m = 1. E, F) Subtractive inhibition (Equation 4.7) with d = 0.15 and m = 1. For graphical notation see Table 5.1 p. 58.

For subtractive inhibition on the other hand there is no gain change at all, the curve keeps it’s shape but is only shifted along the horizontal axis. This follows straight from the definition since we are subtracting a constant from the independent variable of a function.

If we consider the average output from the network for the normalization model, it approaches a constant value $1/n_{mc} \cdot R_{max}$ for increased input to the studied minicolumn. For output gain the average will first increase and then decrease and is thus bounded but not stable. For subtractive inhibition on the other hand the average is strictly increasing and only stable when all non-silenced input units have reached their max firing rate. All these features are seen in Figure 4.4 and a mathematical analysis that shows the same is indeed true also for other values of the constants in Equations 4.5 - 4.7 was conducted and is found in Appendix 7.1.

FIR test theoretical models

If we consider the FIR test, we see that normalization is the only modulatory effect that can keep *both the input relations and the average output* stable for increasing input magnitude while *keeping distinction between the outputs*. These features are all important for the networks information processing abilities e. g. if we wish to see the outputs from the network as encoding probability assessments. Simply put if the inputs are considered as evidence for a certain class, we do not want the probabilities for all outcomes to increase to twice the previous value, if all evidence are increased by a factor of two. This would mean the total probability would no longer sum up to one. It would also not be good if the average output is kept constant but for increased input magnitude the relations between the outputs change. That would mean there would be a different outcome if all inputs were increased by a common factor. Note that while both subtractive inhibition and output gain do have a range for which there is a clear distinction between the outputs, and for subtractive inhibition for low input magnitudes the relation between the outputs is almost stable; in this range the average is strictly increasing. However it is possible to keep the output approximately constant over a limited range of input magnitudes for the subtractive inhibition model, if there is a "starting input" to the minicolumn not creating any modulatory effect as shown in Figure 4.5.

4.3.3 Quantifying the Fixed input relations test

As seen in the previous section a normalizing network, have several important features closely related to its information processing abilities. This was used to quantify the FIR test to enable estimation of if a specific model version can be considered approximately normalizing for some specific input magnitude range and input relations. The features seen in a normalizing network used when designing the quantified FIR test were:

I The *average output* is kept constant independently of input magnitude and the relationships between the inputs.

II The *relations* between different outputs (measured as each outputs relation to the average output) is constant for increased input magnitude.

III There is *distinction* between outputs if there is a clear difference between the inputs. Depending on the value of n in a normalizing network, if and how much the differences between the inputs are magnified in the output differs, but they will not decrease if n >=1.

How these features are quantified is shown in Table 4.5. To evaluate the networks normalizing ability for some specific input relations a sequence of input vectors $\overline{x}_j = a_j\, [c_1, c_2, .. c_n]$ $\sum_i c_i = 1$ is created for a set of logarithmically spaced input magnitudes. For all FIR tests in this thesis the percentual increase in magnitude between two consecutive input vectors is 15 %. The logarithmic spacing means the test result will be independent of scaling the input or changing the input EPSP and that it is not crucial exactly when the normalization starts up.

To calculate the region of approximate normalization for some specific input relations, all input vectors are processed by the network to produce corresponding output vectors. To remove some noise the output vector is

Table 4.5: Quantification of the FIR tests. Mathematical definition of the different criteria for an approximately normalizing network and the FIR score for a specific random setup.

Set of input vectors

$$V = \{ \bar{x_j} = a_j[c_1, c_2, ..c_n] \mid a_j = a_{min} \cdot 1.15^j \wedge a_j < a_{max} \cdot 1.15\}$$

Quantified criteria for an approximately normalizing range

I) Constant average. For any input vector in the range, the average activity in the hypercolumn must not deviate more than p_{mean} percent from the average activity in the hypercolumn *over the entire sequence*.

$$\frac{\sum_i \frac{y_i(\bar{x_j})}{n_{mc}}}{(\sum_j \sum_i \frac{y_i(\bar{x_j})}{n_{mc}})/seq\_length} < p_{mean} \ \forall \bar{x}_j \tag{4.9}$$

II) Distinction between outputs. For any input vector in the range, relations between consecutive entries in the output is at least r_i. (The inputs are assumed to be placed in ascending order in the input vector).

$$\frac{y_i(\bar{x_j})}{y_{i+1}(\bar{x}_j)} < r_i \quad \forall \quad \bar{x}_j \tag{4.10}$$

III) Constant relations between outputs. For any input vector in the range, the output from each minicolumn y_i in relation to the average output, must not deviate more than p_i percent from the average output for that minicolumn over the entire range in relation to the hypercolumn average over the same range.

$$\frac{y_i(\bar{x}_j)/(\sum_i y_i(\bar{x}_j)/n_{mc})}{\sum_j(\sum_i y_i(\bar{x}_j)/n_{mc})/seq\_length} < p_i \quad \forall \bar{x}_j \tag{4.11}$$

FIR score

$$max(\mathrm{a}_{j\_min}/ a_{j\_max}) \tag{4.12}$$

Where a_{j_min} and a_{j_max} is the average input for the first and last vectors in a set of consequtive vectors that fulfill Equation 4.9 – 4.11 the average input for the first vector in the range, divided by the average input for the largest input vector

Parameters for the quantified FIR tests

	$\bar{c}$	P_{mean}	$\bar{r}$	$\bar{p}$
1234 test	[0.1 0.2 0.3 0.4]	0.2	[0.5 025 0.15]	[F 0.25 0.25 0.25]
1234 average test	[0.1 0.2 0.3 0.4]	0.2	[F F F]	[F F F F]
1200 test	[1/3, 2/3, 0, 0]	0.2	[0.25] y1 > 0.1y_{mean}*	[0.25 .25]
1200 average test	[1/3, 2/3, 0, 0]	0.2	[F F F]	[F F F F]

F = free (no condition applied) * an additional condition of the smaller output being at least 20 % of the average output.

averaged over four runs of the specific network setup. The quantified criteria detailed in Table 4.5 are then used to evaluate the sequence of output vectors to give the longest region of input magnitudes over which the network is considered approximately normalizing (for the specified input relations). The FIR score for a specific random setup is then defined as the quotient between the smallest and the largest input magnitude in this sequence of input vectors. This range of input magnitudes is also shaded in the FIR graphs. When comparing different network model versions, the FIR score is averaged over five different setups using different random seeds. This is done to give a fair comparison between model versions since the score usually differ between setups.

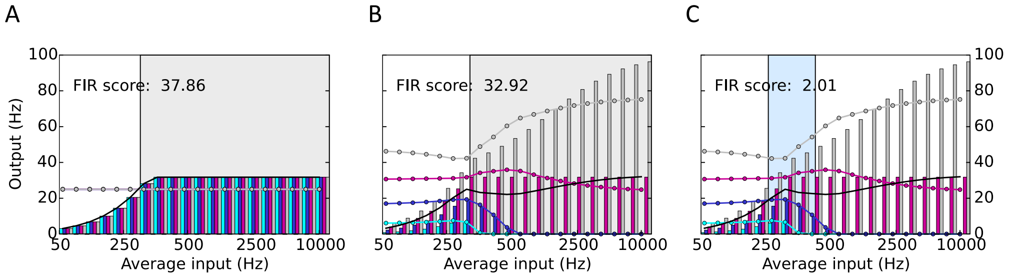

Figure 4.5: Keeping average constant with subtractive inhibition. All plots use Equation 4.7 to simulate network activity with R_{max} = 100, σ = 500, and n = 1.5, but subtract a constant m = 300 from the independent variable in g($\overline{x}$). A) FIR average test with inputs $\overline{x}$ = a[0.25, 0.25, 0.25, 0.25]. If we have a starting input before the modulatory effect starts up, it is trivial to keep a stable average when all minicolumns have the same input. Here k = 0.25. B) FIR1234 average test. When there is different inputs to the minicolumns keeping average is more complicated. However by letting some outputs increase and other decrease the average can be kept approximately stable over a large range. C) FIR1234 test. For the inhibition strength that keeps average constant, the relationships between the outputs is diverging, which results in a low FIR score. For graphical notation see Table 5.1 p. 58.

A graphic illustration of evaluating the theoretical models with the quantified FIR tests is seen in Figure 4.6. The results show that these tests indeed favor the normalization model, and give poor results for the other models, depending on their bad ability to a) keep average and relations between outputs constant and/or b) keep distinction between outputs. It should however be noted that a network can pass these tests for a certain range even if the underlying equations describing the network is not the normalization model. A single FIR tests can thus *not be used to prove* that a network is normalizing, which is also not the objective for this test. For evaluating the networks model versions it is as important to look at qualitative features in the FIR graph, such as if average is strictly increasing or how the relationships between outputs from different minicolumns change. The FIR test is also designed to be useful for studying when and why normalization starts up and breaks down. The score from the quantified FIR tests are used to compare different model versions to see if certain changes tend to improve normalization or not. This should not be seen as definite proof that one network normalizes better, since other factors need to be taken into account as well, but it is a strong indication of better normalization.

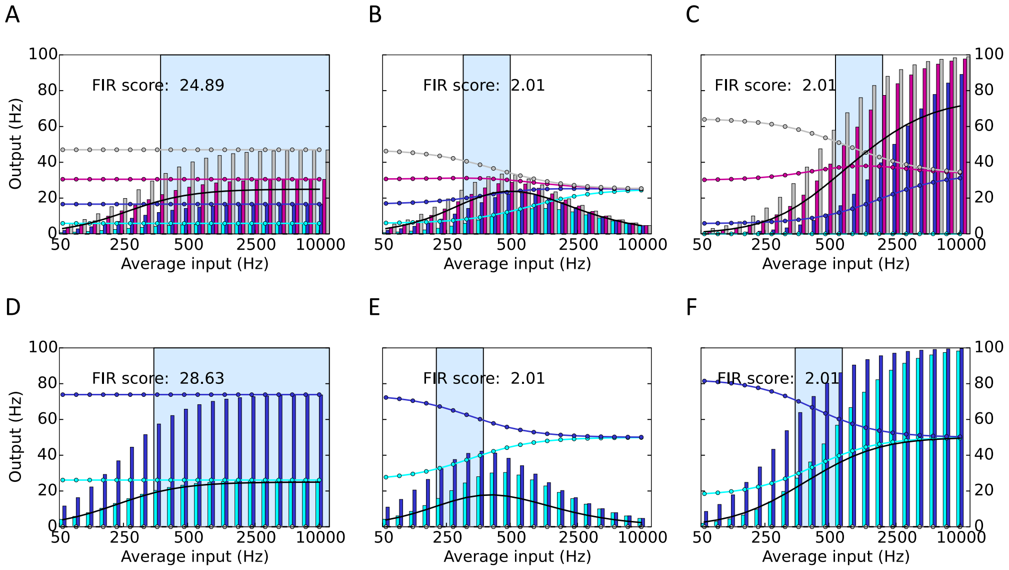

Figure 4.6. Result of applying the quantified FIR tests to the theoretical models. First row FIR1234 test, second row FIR1200 test. A, C) Normalization. B, E) Output gain. C, F) Subtractive inhibition. Equations in Table 4.4 are used to calculate the output for the set of input vectors specified in Table 4.5, with n = 1.5, R_{max} = 100 and σ = 500. For normalization d = 1.0, for output gain d = 0.0005 and for subtractive inhibition d = 0.15. For graphical notation see Table 5.1 p. 58.

5 Results

This chapter first presents some results on how the IO curve of a single model neuron and minicolumn is affected by increased amounts of Poisson inhibition. Then the results from Model A with standard parameters are shown. Thereafter the effects of feed-forward and feed-back inhibition only, that is Model A1 and A2, are studied, to help understand network functionality, and find eventual mechanisms that contribute to a normalizing network. Then Model A tuned for improved performance is revisited and a number of additional changes are evaluated. Those include increased variability in the network; changed strength of the recurrent connectivity in the pyramidal and basket cell populations; and the use of model neurons with more realistic firing characteristics. Last Model B and the effect of adding short-term depression is evaluated. The chapter is organized to let the reader follow the investigation. Since interpretation of the results were important for choosing the next step, there is also a discussion of implications and causes for different results. The Discussion chapter however give a more comprehensive discussion of results and implications.

Notation for IO and FIR graphs

Since a large number of similar graphs is presented in the results chapter, the graphical notation for the IO and FIR graphs, is not repeated in the subtext of each graph but instead summarized in Table 5.1 and Figure 5.1.

5.1 Model A standard parameters

This section presents results from studying a single neuron, a single minicolumn and Model A with standard parameters. These results constitute a starting point for further investigations of the hypercolumns normalizing ability. For all experiments in this section the standard parameters for Model A found in Table 4.2 were used.

5.1.1 Single model neuron and minicolumn

First the effect on the model neuron response curves when adding a small current or conductance based inhibitory input was studied. This was then repeated for a single model minicolumn receiving inhibitory input from the basket cell population. The purpose of these tests were to see if different modulatory effects are present in a single neuron, a single minicolumn and the complete hypercolumn module.

IO curves for model neurons

How the model neurons response curves are affected by different amounts of direct inhibition (as opposed to inhibition mediated through the basket cell population) is seen in Figure 5.2. The pyramidal cell response curve is a good fit to a sigmoidal Hill function (Equation 4.4) with $R_{max} = 285$, $n \approx 1$, $\sigma \approx 1500$ and increased offset β for larger amounts of inhibition. The basket cell response curve have similar fit parameters but a higher maximum output. The clear horizontal shift of the curves for increased amounts of inhibition, show that the modulatory effect of an inhibitory input, is best described as subtractive. This applies to both the pyramidal and basket cell, and implies that with this neuron and synapse model and these synaptic weights, the model neurons does not *intrinsically* implement a mechanism that result in normalization or output gain control, when receiving a inhibitory synaptic input. A feature not explained by a pure subtractive shift of the IO curve is that the onset of firing is smoother for a neuron receiving Poisson inhibition. This means the shape of the response curve is changed in a way that reduce gain for low firing rates.

Table 5.1: Summary of graphical notation for FIR and IO graphs

Notation IO test graphs	
Continuous lines	Activity in studied minicolumn.
Dotted lines	Activity in the minicolumn that receives the second largest input of the not studied minicolumns.
Dashed lines	Average activity in the hypercolumn.
Color code	Lines of the same color correspond to a specific modulatory input. For sigmoid fits only the activity in the studied minicolumn is seen.

Notation FIR test graphs	
Continuous black line	Average activity in hypercolumn.
Dashed black line	Activity in basket cell population.
Histograms	Each group of bars show activity in the network minicolumns (that is network output) for a specific input magnitude. Bars of the same color correspond to the same minicolumn.
Continuous lines with dots	Activity in each minicolumn as percentage of the average activity in the hypercolumn. Color coded to correspond to the minicolumns in the histogram. This is plotted against the scale of the y-axis as percentage.
Blue shaded region	Longest range passing the FIR1234 or FIR1200 test.
Grey shaded region	Longest range passing the FIR1234 average or FIR1200 average test.

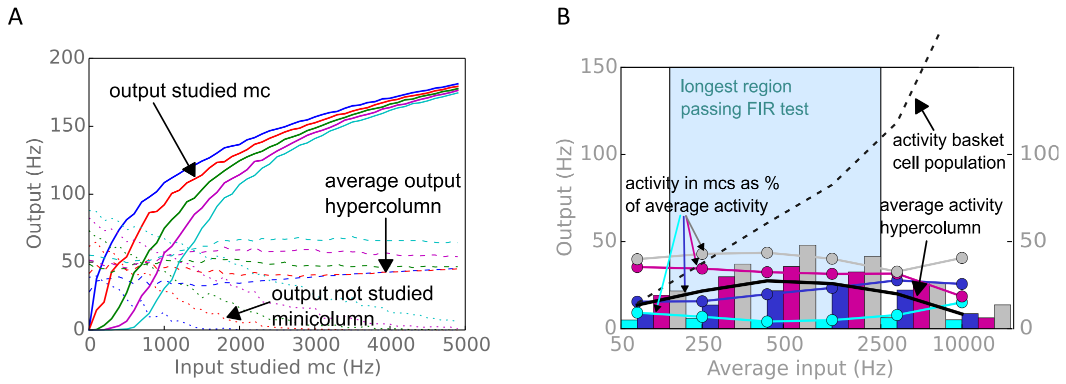

Figure 5.1: Notation for IO and FIR graphs. A) IO graph. B) FIR graph.

This is a feature typical for an in vivo neuron experiencing a noisy input (Silver 2010). Here the increased variance is a result of that a) For higher amounts of inhibition a higher input frequency is needed to reach a certain output and the variance of the driving Poisson process will thus be higher (for a Poisson process variance equals average rate) and b) The inhibition will also contribute to the variance since the variance of the superposition of two independent subprocesses (here the excitatory and inhibitory Poisson inputs), equals the sum of the individual variances.

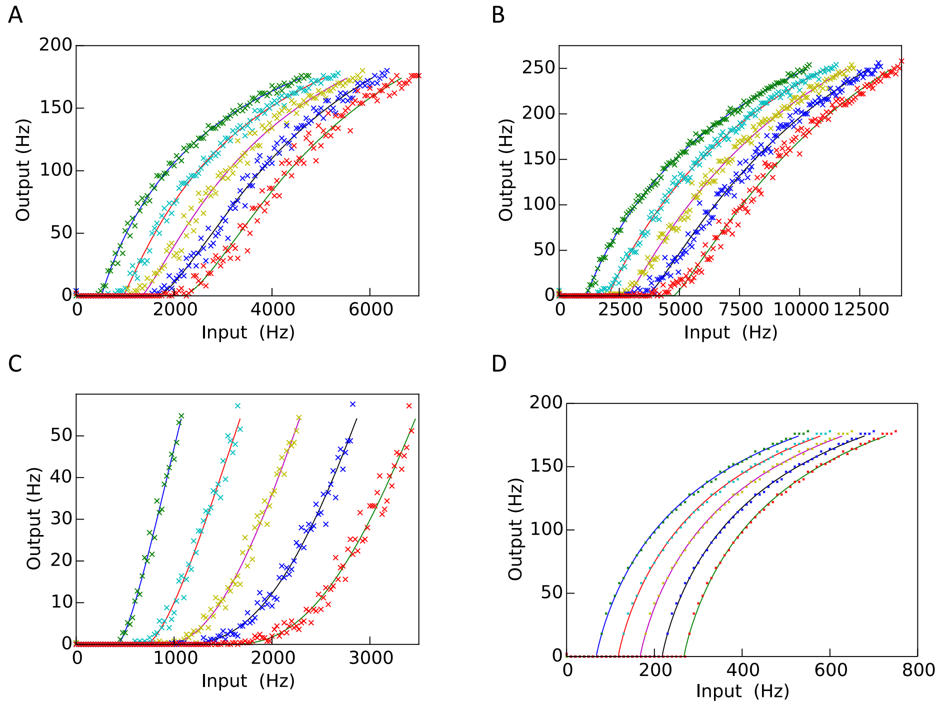

Figure 5.2: How the model neurons IO curves are affected by increased amounts of inhibition. All lines are fits to a sigmoidal Hill function (Equation 4.4). A) Pyramidal cell receiving Poisson excitation and inhibition. Inhibition: 0, 200, 400, 600 and 800 Hz. Fit is done to data points below 175 Hz output. Note the clear subtractive shift of the IO curve. B) Same as A, but for basket cell. C) Same as A, but the fit is done to points below 50 Hz output. Note how increased Poisson input and inhibition creates a smoother onset at rheobase which makes the IO curve similar to a power function at the firing onset. The dominating feature is however still the subtractive shift. D) Pyramidal cell instead driven and inhibited by a steady state current. Inhibition: 0, 50, 100, 150 and 200 pA. Note the sharp onset of firing and that the curves are perfect subtractively shifted versions of each other.

Single minicolumn receiving basket cell inhibition

To test how a single minicolumn is affected by increased amounts of inhibition the parameters and setup from Model A were used. This means the same noise input to pyramidal cells, input variance, recurrent connectivity within the minicolumn and connectivity from basket cell population to the minicolumn. However the input to the basket cells was for this test *not related to the input or the activity in the minicolumns* (as is the case in the hypercolumn module). That is for this test the network was ran with only one active minicolumn, and the input to the basket population varied freely.

The result is seen in Figure 5.3. Not surprisingly the IO curve for a single minicolumn is still well described by a sigmoidal Hill function. When a modulatory input is added the IO curve is shifted to the right on the horizontal axis, while gaining a smoother threshold at rheobase. This is in analogue with how the IO curve of the single neuron was affected, and for these parameter values using a computational unit composed of several neurons does not result in any qualitatively different effect. One difference is that output from a minicolumn is less noisy than that from a single neuron because averaging reduce variance. The IO curve also has a steeper slope, which is a consequence of the recurrent connectivity in the minicolumn.

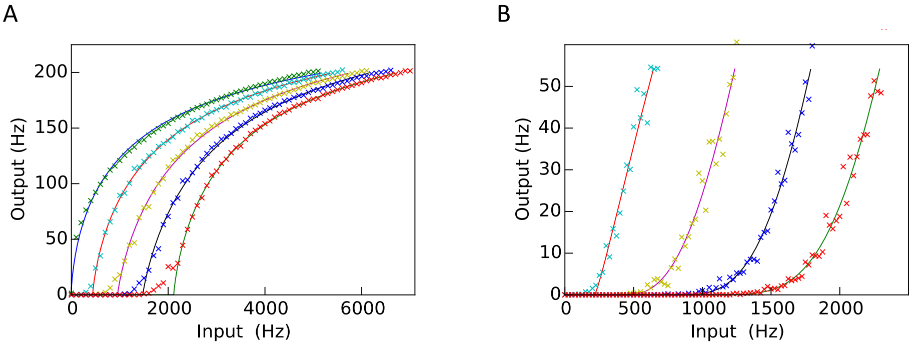

Figure 5.3: The effect on the IO curve of a single minicolumn of increased amounts of inhibition from the basket cell population. Lines are Hill function fits (Equation 4.4). A) IO curves for a single minicolumn. Driving input to the basket cell population: 0, 1400, 1700, 2000 and 2300 Hz. Fit is done to outputs below 200 Hz. Note that increased inhibition does not affect the max firing rate, only when it is reached. B) Same as A but the fit is done to points below 55 Hz.

5.1.2 No inhibition

For reference the networks information processing characteristics without any inhibition was studied. This means Model A, but with a connection probability of zero between basket and pyramidal cell populations as well as between pyramidal and basket cell populations. The result of the IO test and the FIR1234 test are seen in Figure 5.4 A and B. Response curves for different amounts of input to the other minicolumns all collapse to a single curve since without feed-back or feed-forward inhibition the input or activity in the other minicolumns do (of course) not affect the studied minicolumn. The average is not surprisingly strictly increasing for increased input to the studied minicolumn, as well as for increased input to the other minicolumns.

The FIR1234 test show one feature which tend to give this setup poor information processing characteristics. As the input magnitude increase the differences between the outputs diminish and thus gradually any information about the relationships between the inputs is lost, as all units reach their max firing rate. This non inhibitory network pass the FIR1234 average test over a small region simply because a certain increase in average is allowed. It does however not pass the FIR1234 or FIR1200 test for any range which is because of it’s poor ability to keep distinction between outputs. This is also illustrated in the spike histogram and raster plot for an input vector $\overline{x}$ = [200, 400, 600, 800] Hz seen in Figure 5.4 C and D. The outputs are clearly more similar in magnitude than the inputs even for this quite moderate input magnitude. If this result is compared to the network model versions with inhibition, independently of their specific modulatory effect, it is clear why it can be advantageous for the biological nervous system to "step at the gas and the break simultaneously" (Isaacson & Scanziani 2011).

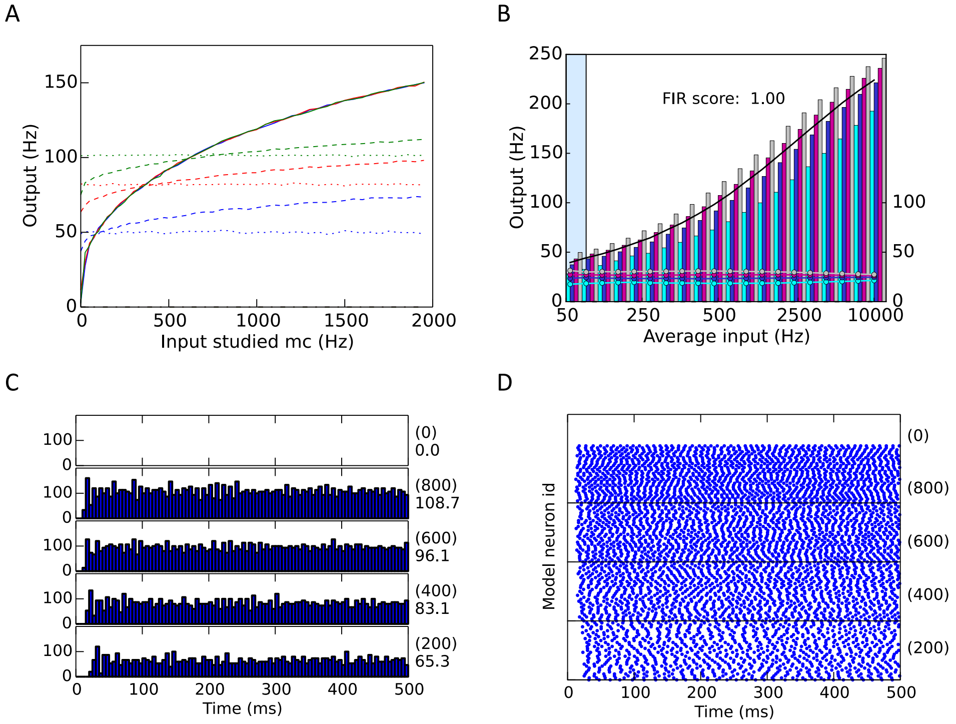

Figure 5.4: Characteristics of Model A standard parameters, without inhibition. A) IO test. Input to the other minicolumns: 100, 400 and 700 Hz for blue, red and green curves respectively. Note that all IO curves for the studied minicolumn coincide. B) FIR1234 average test. We see that average is strictly increasing and that the relations between the outputs is diminishing for increasing input magnitude. C) Spike histogram for the four minicolumns, for input $\overline{x}$ = [200, 400, 600, 800] Hz. D) As C but raster plot. For C and D numbers in parenthesis to the right of each figure are input to the respective minicolumns and the basket cell population. In figure D also average activity in each minicolumn and the basket cell population is seen.

5.1.3 Model A – complete network

As a starting point for further investigations, Model A with standard parameters was studied. Some network characteristics for two input vectors $\overline{x}_1$ = [200, 400, 600, 800] Hz and $\overline{x}_2$ = [800, 1600, 2400, 3200] Hz are seen in Figure 5.5 and Figure 5.6. It can be seen that all minicolumns are suppressed by the activity in the basket cell population, and that the distinction between outputs is better compared to the network without inhibition. It can also be noted that there is a short delay before the inhibition sets in – the same effect is seen in cortex (Isaacson & Scanziani 2011) - and thus all minicolumns start up firing, but as soon as the basket cell population gets active the activity is dampened and the minicolumns receiving a smaller input is almost or completely silenced.

After 5 to 25 ms the network is stable in the sense that there is either oscillatory activity with a fixed frequency and amplitude, or more irregular firing. For both cases there is a stable output rate if averaged over approximately 100 ms. For the lower magnitude input vector in Figure 5.5 there is a pronounced oscillatory activity with frequency determined by the synaptic delay time and the membrane time constant. This oscillatory activity is created by the dynamics of the feed-back inhibition and here gives gamma oscillations. Gamma oscillations was also seen in the original attractor memory model (Lundqvist et al. 2006). That oscillatory activity is more pronounced for lower magnitude inputs depends on that feed-back input to the basket cells then dominate over the feed-forward input.

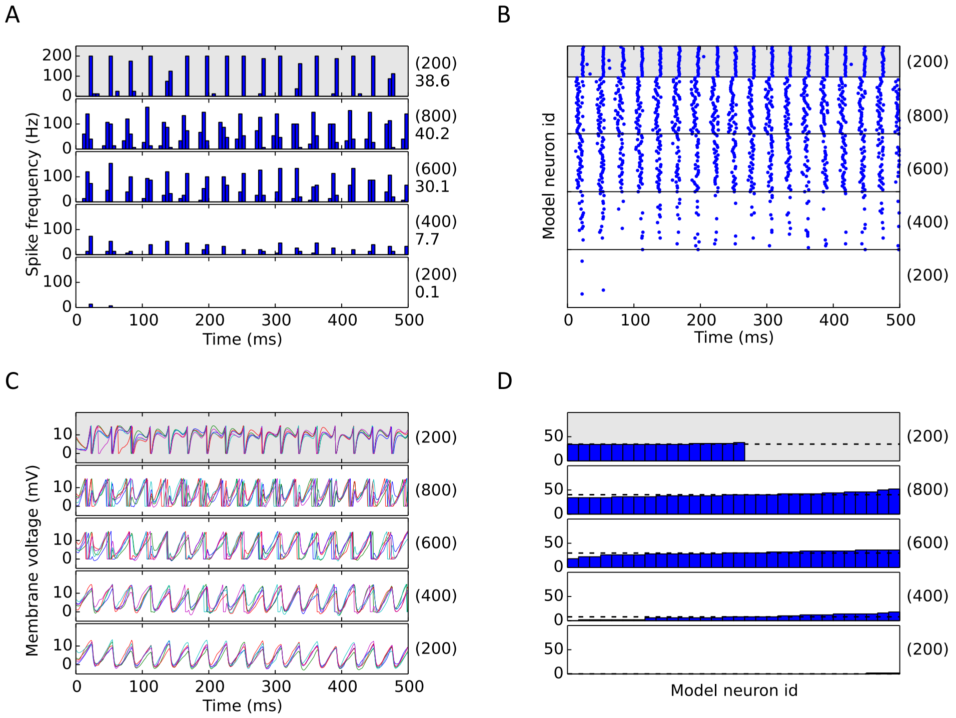

Figure 5.5: Network activity for a 500 ms run with input $\overline{x}$ = [200, 400, 600, 800] Hz. A) Spike histogram. Note that the minicolumns with smaller input start up spiking but are silenced after a couple of milliseconds. B) Raster plot. C) Voltage traces for 5 randomly chosen neurons in each minicolumn. D) Distribution of average spike frequency for neurons in each minicolumn. Neurons are sorted according to their average firing rate and dashed line is average activity in minicolumn. Numbers to the right of each figure in parenthesis show input to the respective minicolumns. Numbers below those in A show average firing rate in each neural population.

That single cells reach higher output rates than in the original model (especially pronounced when there is oscillatory activity) is a consequence of the absence of short-term synaptic depression and adaptation and that the driving input to a minicolumn can be considerably stronger than in the attractor memory model. The output from the network show clear signs of average control. The average activity in the minicolumns (and thus also the hypercolumn) is although slightly larger for Figure 5.6 clearly smaller than twice the average for the smaller input vector even though input magnitude was increased by a factor of four. It should also be noted that, for both input vectors, minicolumns with smaller inputs, are suppressed to a higher degree than those with larger inputs. When studying the distribution of average spike frequencies for each minicolumn it can be seen that the randomization of neuron size and driving input creates a small variance in firing frequency.

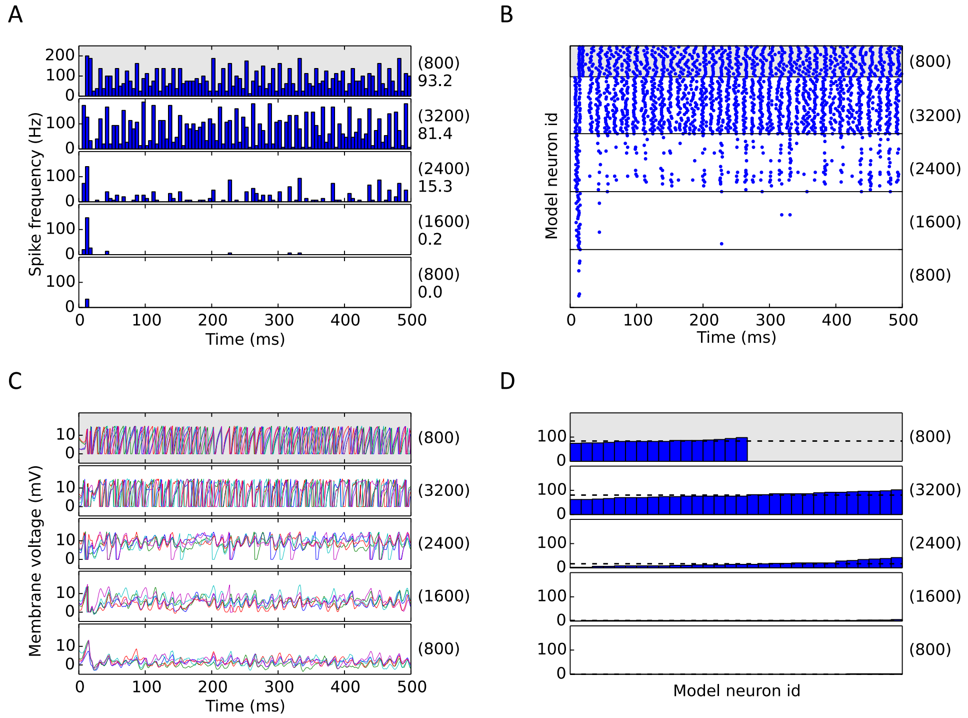

Figure 5.6: Network activity for a 500 ms second run with input $\bar{x}$ = [800, 1600, 2400, 3200] Hz A) Spike histogram. B) Raster plot. C) Voltage traces for 5 randomly chosen neurons in each minicolumn. D) Average spike frequency for neurons in respective minicolumn. Dashed line is average activity in minicolumn.

IO test Model A standard parameters

The result of IO test for Model A with standard parameters is seen in Figure 5.7 and Figure 5.8. To get the bigger picture the network was first tested on different input magnitude scales. In Figure 5.7 A two regions can be discerned in the IO curve: First the output start saturating at a level lower than the max firing rate given by the neuronal parameters, but then the average starts increasing again approaching the same max firing rate seen for the network without inhibition. In the first region (in the following referred to as “the modulatory region”) it is clear that the activity in the studied minicolumn is suppressed by increased input to the other minicolumns as well as the other way around. However, in the second region all curves almost collapse to one, indicating that we no longer have any/very small modulatory effects on the studied minicolumn. This depends on that the input to the studied minicolumns is so much larger than the input to the other minicolumns. For higher inputs we can also note that the hypercolumn average is ¼ of the average firing rate of the studied mc – this since all the other minicolumns are silenced - and that the average keep increasing until the output of the studied mc saturate.

So in the modulatory region the feed-forward and feed-back inhibition do keep the output approximately bounded over a certain range of input magnitudes, but as input magnitude increases this is no longer the case. In fact for the setup used in all model versions in this thesis, the average control will always break down eventually since the excitatory input can be increased without bound, while the inhibition from the basket cell population is bounded by $EPSC_{pyr\_bas} \cdot N_I$.

$R_{max\_bas} \cdot p_{bas\_pyr}$. Inhibitory activity will also grow sublinearly for a linear increase in excitatory input for a range before that happens.

However since a biological system always have an operating range (and the input to a biological minicolumn in vivo will be bounded), it was not considered as crucial whether the system breaks down for very high input magnitudes or not. Instead we were primarily interested in the region of lower input showing the modulatory effects we wish to study. However since one of the benefits of normalization is to extend the dynamic range of a system, it was seen as an advantage with network model versions that prolong the operating range.

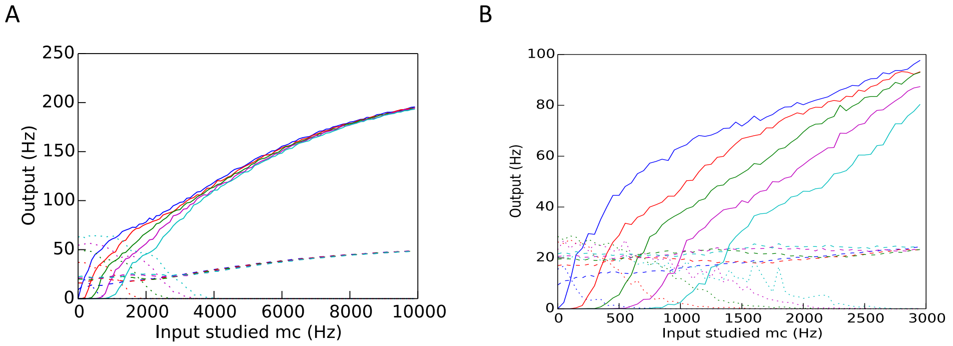

Figure 5.7: Result of IO test for Model A on different input magnitude scales. Average input to additional minicolumns: 100, 400, 700, 1000 and 1300 Hz for both plots. A) Input to studied minicolumn 0-10 000 Hz. B) Input to studied minicolumn: 0-3000 Hz. See table 5.1 for graphical notation.

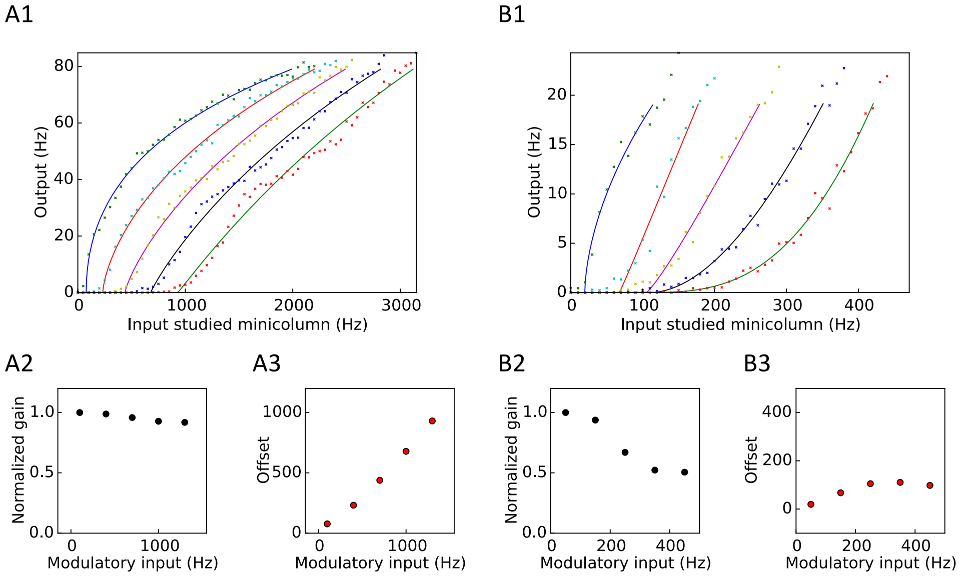

Figure 5.8: Hill function fits (Equation 4.4) for Model A, standard variables. A1) Hill function fit using datapoints below 80 Hz output. A2. Modulatory inputs: 100, 400, 700, 1000 and 1300 Hz. A2) Normalized gain for curves in A1 between 5 and 80 Hz output. A3) Offset for curves in A1. B1) Hill function fit to data points below 20 Hz. Modulatory input: 50, 150, 250, 350 and 450 Hz. B2) Normalized gain for curves in B1 calculated between 2.5 and 20 Hz. B3) Offset for curves in B1.

The results from the IO test in the modulatory region is studied in more detail in Figure 5.8 where a sigmoidal Hill function (Equation 4.4) is fitted to the data from Figure 5.7 B as well as to data from performing the IO test with smaller modulatory inputs. To visualize the amount of gain change “normalized gain” were computed for the fitted functions. This is as described in Section 4.3.1 calculated as the average derivative for each curve between some chosen y_{min} and y_{max} divided by the average derivative for the curve corresponding to the smallest modulatory input. Offset were also plotted as function of the modulatory input, where offset is defined as the subtractive shift β in Equation 4.4. It is clear that for larger modulatory inputs and larger output rates the effect on the response curve of the studied minicolumn is primarily subtractive. There is in principle no gain change, and a very pronounced subtractive shift to the right on the horizontal axis.

However as seen in Figure 5.8 B1-B3, for smaller modulatory inputs and low output rates there is in addition to a shift in rheobase or “offset”, also a noticeable gain change. This is even more pronounced for very low output rates (below 5 Hz). For input rates over 10 Hz the subtractive shift dominates also here.

FIR tests

The results of the FIR1234 and FIR1200 tests are seen in Figure 5.9. It is clear that for high input magnitudes the FIR characteristics are similar to the subtractive inhibition model. The distinction between the outputs gradually diminish as they reach their max firing rate for increased input magnitude while the average output from the hypercolumn is increasing fast. However in the low input region there is better distinction between the outputs compared to the network without inhibition. At the same time the output relations are stable over a larger range of input magnitudes than what is the case for the subtractive inhibition model (compare to Figure 4.4 F and 4.5 C). This indicate that the network show some weak normalizing effects. The longest approximately normalizing region according to the FIR tests lies between 150 and 500 Hz input magnitude. This is in agreement with that it was for this region a gain change was seen for the IO test.

Considering the relationships between the outputs (remember continuous lines with dots plot the output in each minicolumn as fraction of the average output), a “bubble” is seen: First the differences between the outputs diminish since the inhibition is not yet strong enough to counteract the increasing input magnitude, but then the differences start increasing and the minicolumns with smaller inputs are almost completely silenced - the network enters a WTA mode. After that the distinction between the outputs is gradually lost as the inhibition no longer keeps up with the increasing input magnitude.

This means the network suppresses smaller inputs to a higher percentage for medium input magnitudes and that input magnitude thus has a substantial effect on the network functionality, which is not the case for a normalizing network. This first half of the bubble can though be compared to Figure 4.5 C and the second half to Figure 4.6 C, which give at hand that these features arise in a system characterized by subtractive inhibition. It could be noted that the longest region of approximately stable output relations is seen where there is a slowly increasing average, and that the region of approximately stable average is considerably longer than the region of approximately stable output relations, but at the far end exhibiting WTA features.

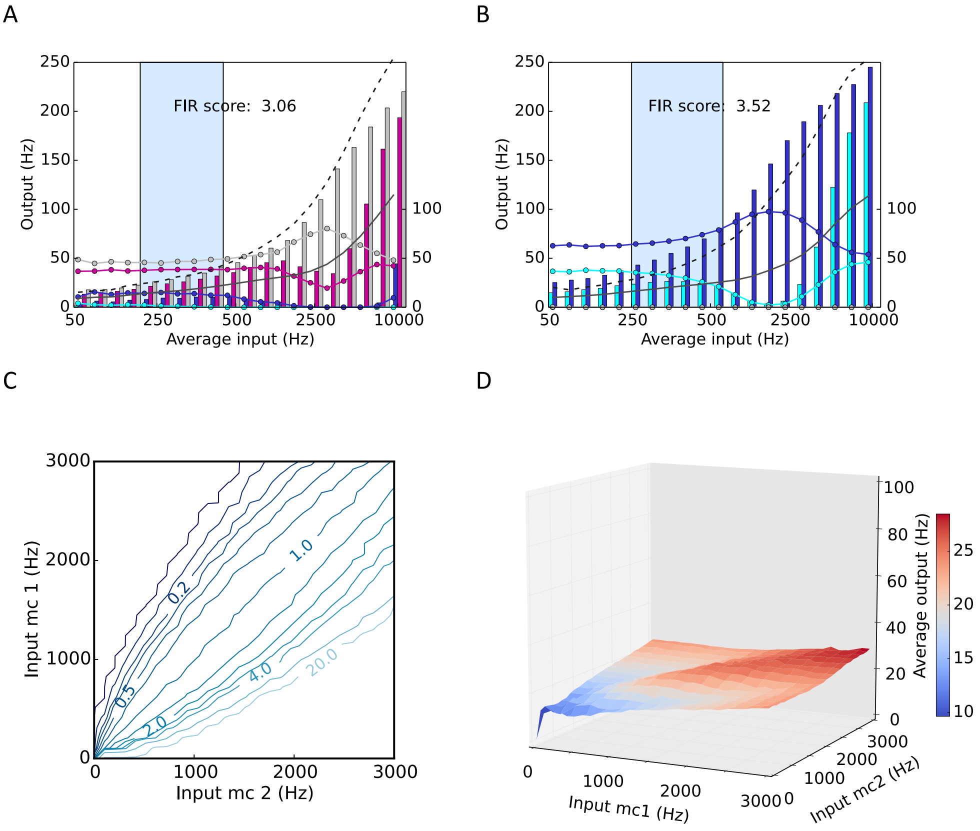

Figure 5.9: Some network characteristics for Model A standard parameters. A) FIR1234 test. B) FIR 1200 test. Note the "bubble" where the relationships between the outputs first increase and then decrease. C) Relationship between outputs for a network with two active minicolumns, as a function of input to the respective minicolumns. D) As C but average activity in the network as a function of input to the two minicolumns. Note that average is neither stable for increasing input magnitude, nor for fixed input magnitude and varying input relations.

That the qualitative features of the two different FIR tests are similar indicate that both can be used to estimate the normalizing ability of a network. To further test this hypothesis over a larger range of input relations, a second test was performed were the relation between the outputs for a network with two active minicolumns, as a function of the input to the respective minicolumns was studied. That is the output from the network $\overline{y}$= [y_1, y_2, 0, 0] was evaluated for each data point and the graph depicts the contour plot of y_1/y_2. The result is seen in Figure 5.9 C. Note that for a specific relationship between the inputs, the output for increased input magnitude can be traced by following a straight line in the graph with slope decided by the relationship between the inputs. If such a line cross the contour lines of the plot this means the relationship between the outputs change with input magnitude.

That this is clearly the case for this network independently of the slope of the line (as long as it is not one), show that the characteristics seen in the FIR graphs is true also for other input relations. It should however be noted that the change in output relations happens faster for more dissimilar inputs. This is important since it means to prove that a network is normalizing in a certain range of input magnitudes, it is not enough to study a single FIR test; it has to be combined with this type of test and the IO test.

IO test vs FIR test

How can the results from the IO test and the FIR tests connected? It should be noted that for the IO test in contrast to the FIR test, the relationship between the inputs is not constant for increased input magnitude but a certain relationship between the inputs is tied to a certain input magnitude. The quotient between the input to the studied minicolumn and input to the other minicolumns will thus for low input magnitudes be very low and for high input magnitudes very high. Since the network tend to suppress an input that is small in relation to the average input to a higher degree than larger inputs, this explains why also for the region where the network can keep output relations stable for the FIR1234 test, the IO curve does not follow the normalization model (even if some gain change is seen).

Concluding remarks Model A standard parameters

Model A with standard parameters show weak normalizing features for low input magnitudes and inputs similar in input magnitude. Within a range of 100-500 Hz the input magnitude can be increased with factor of three while keeping approximately the same average activity and relationships between outputs (where approximately means there in fact might be about a 50 % increase). The result of the IO test and the qualitative features of the FIR tests however show that the network, although able to approximate normalization over a certain range of input relations and magnitudes, is qualitatively most accurately described by the subtractive inhibition model. However some normalizing features are seen which are not explained by this model so the next step in the investigation was to try and understand what creates those and whether the mechanisms might be possible to extend. As a basis for further investigations a hypothesis of the parameters most likely to affect the modulatory effects of the network was constructed.

5.1.4 Hypothesis of important parameters

To evaluate the hypercolumn module the effects of different parameter changes needed to be evaluated. It is not feasible to systematically test all possible parameter combinations for the vast parameter space of a spiking neural network model. The question is then which parameters changes should be evaluated? One way of reducing the parameter space is considering dependencies, for example is it probable that the PSP sizes are more important than the specific neuronal membrane parameters and synaptic weights.

The choice of parameters and features to investigate was primarily based on that the fact that the network is designed for population rate coding. Since the output rate for a certain input is decided by a computational units IO curve, the most important parameters are those that affect either the *strength* of the excitatory or inhibitory drive to the different neuron populations or the *shape* of the IO curve. This means for example the fraction of the input to the minicolumns that is fed to the basket cells as feed forward inhibition and the strength of the feed-back from the minicolumns to the basket cells. It also includes, the strength of the inhibition from the basket cells to the minicolumns as well as, the recurrent connectivity within the basket cell and pyramidal cell population.

For further investigations the starting point was the hypothesis that the three most important parameters for network functionality is (see Table 4.2. for parameter definitions):

- Strength of the feed-forward drive to the basket cell population:

$$\propto p_{in\_bas} \cdot N_I \cdot EPSC_{ext\_bas} \quad (5.1)$$

- Strength of excitatory drive from the minicolumns to the basket cell population:

$$\propto EPSC_{pyr\_bas} \cdot N_E \cdot p_{pyr\_bas} \quad (5.2)$$

- Strength of inhibitive drive from basket cell population to minicolumns:

$$\propto EPSC_{pyr\_bas} \cdot N_I \cdot p_{bas\_pyr} \quad (5.3)$$

- Noise input to the pyramidal cells in relation to noise input to the basket cells. Since noise input to pyramidal cells will be held constant, this reduce to:

$$noise_{bas}$$

The strategy when tuning the different network versions was to start with inhibition strength and noise to basket cells to create a network that performed good on the IO test as well as the FIR1234 and FIR1200 tests. For the IO test this in practice means gain change and average control, and for the FIR test average control and stable output relations for as long input range as possible. Changing pyramidal and basket cell parameters, such as average size and membrane time constants, was not tested explicitly for different model versions. However during some exploratory studies in the beginning of the project those were not found to affect the network behavior in any significant way as long as synaptic weights were tuned to give similar PSPs.

When a network model version was tuned, for some version additional parameters were tested, and. Those include primarily:

- Recurrent connectivity in basket and pyramidal cell population

$$p_{pyr\_pyr} , p_{bas\_bas}$$

- Standard deviation for input to pyramidal and basket cells:

$$Input_{rsd\_pyr} , Input_{rsd\_bas}$$

- Standard deviation on pyramidal and basket C_m.

$$C_{m\_rsd\_pyr} , C_{m\_rsd\_bas}$$

A feature not captured by any specific parameter, that were also investigated was using a variable number of incoming connections to postsynaptic neurons.

5.2 Model A1 - Feed-forward inhibition

This section describes the result from studying Model A1, that is Model A without any feed-back inhibition. The aim of the performed tests was to discern which role feed-forward inhibition plays in creating the weak normalizing effect seen in the low input magnitude region for Model A standard. First the network was studied with the same standard parameters used in Model A. Then the effects of different parameters changes were investigated. A simplified mathematical model of the feed-forward network was also constructed and analyzed to better understand its modulatory effects.

Network characteristics for a single input vector

First network characteristics for a single 500 ms run of Model A1 with input $\overline{x}$ = [400, 800, 1200, 1600] Hz was studied. The results are seen in Figure 5.10 and are similar to those seen for Model A. However for feed-forward inhibition there is no clear oscillatory activity in the network although there are dependencies between the different minicolumns introduced by their common inhibitory input. The excitatory input to the minicolumns proceeds the inhibitory by a couple of millisecond, a dynamic also present in cortex for this type of feed-forward circuit (Isaacson & Scanziani 2011).

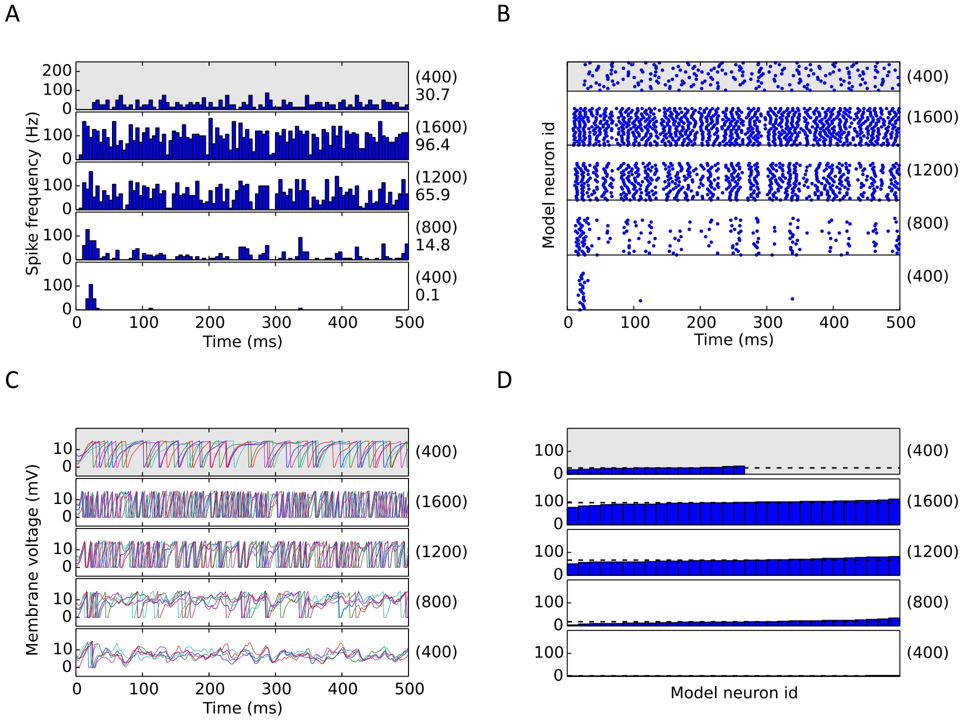

Figure 5.10: Network activity in Model A1 for a single input vector $\overline{x}$ = [400, 800, 1200, 1600] Hz A) Spike histogram. B) Raster plot. C) Voltage traces for 5 randomly chosen neurons in each minicolumn. D) Distribution of spikes for neurons in each minicolumn. Dashed line is average activity in minicolumn. Numbers in parenthesis to the right of each figure show input to the respective minicolumns. In A also average firing rate in each neuron population is seen.

5.2.1 Tuning inhibition strength.

How is the normalizing ability of Model A1 affected by different parameter changes? If we considered the parameters hypothesized to have the largest impact on network functionality, those will for Model A1 be reduce to feed-forward drive to the basket cell population and the strength of the inhibitory drive to the minicolumns. The effect of changing the feed-forward drive was examined by varying both p_{in_bas} and $EPSC_{ext\_bas}$. The effect of changing the inhibitory drive from pyramidal to basket cells was primarily investigated by varying $IPSC_{bas\_pyr}$.

Keeping average constant

One feature of a normalizing network is as mentioned that it keeps the average output constant for increased input magnitude. The intuitively most direct way to accomplish this in a feed-forward network is to tune inhibition strength to counteract any increase in input magnitude. This approach was used to see if it could improve the networks normalizing ability. Both changing the feed-forward drive to the basket cell population and the strength of the inhibitory connections from the basket cell population to the pyramidal cell population was tested, using the FIR1234 as a measurement of the networks normalizing ability. At the same time the FIR1234 average test was used as a measurement of average control.

The change in how the network process an input vector with constant input relations when tuning $\mathrm{EPSC}_{\mathrm{ext\_bas}}$ is seen in Figure 5.11 A1-A3. It is clear that a too small synaptic weight will not give enough average control. A too large value on the other hand will increase the rate at which the relations between the outputs diverge for increasing input magnitudes. Considering the relationships between the outputs, the result are qualitatively similar to those for Model A with standard parameters, creating the same characteristic “bubble” of first increasing and then decreasing relationships between the inputs, although this is less pronounced when inhibition is weaker.

To investigate performance over a more extensive range of parameter values, the FIR1234 test and the FIR average test was performed for 20 logarithmically spaced $\mathrm{EPSC}_{\mathrm{ext\_bas}}$ values in the range [0.0014, 0.17] pS. A slightly smaller range is plotted in Figure 5.11 D to highlight the interesting part of the curve. The FIR scores are averaged over 5 random seeds (note that for the FIR *graphs* each graph is produced using a single random seed, since they reflect the performance of a single random setup). The result of the parameter search show that a quite narrow range gives the best FIR scores. It should also be noted that the best average control does not coincide with the longest normalizing region, although they are quite close in the $\mathrm{EPSC}_{\mathrm{ext\_bas}}$ space. This unfortunately means that simply tuning inhibition strength to keep average constant is not enough to create a normalizing network. The best FIR1234 score lies in a similar range (between 5 and 6 times increase in input magnitude) if instead $p_{\mathrm{in\_bas}}$ or $\mathrm{IPSC}_{\mathrm{bas\_pyr}}$ are tuned (see Figure 5.11 E-F). Tuning $\mathrm{IPSC}_{\mathrm{bas\_pyr}}$ however resulted in the longest region of approximately stable average - almost 30 times increase in input magnitude - but we see WTA behavior and not normalization over a large part of this range. However, it should be noted that the result of tuning these three parameters are qualitatively very similar.

Basket cell IO curve and average control

To understand why there is a limit to the longest region of stable averages it should be remembered that the IO curve of the basket cell population is sigmoidal. This means that a) There is no region of stable growth, but the gain is first increasing and then decreasing (see Figure 5.14 for a plot of the Hill function derivative) and b) Eventually the basket cell curve will start saturating and thus no longer keep up with the growing input. The first property means that for increased input strength the gain of the basket population’s IO curve will first be supralinear and then sublinear (note the logarithmic scale on the horizontal axis for the FIR plots). This is the reason a feed-forward drive to the basket cell population appropriate for a stable average in the lower input region can result in too much inhibition for the medium input region.

It is also clear that a problem when strengthening the inhibition by scaling p_{in_bas} or $EPSC_{ext\_bas}$, is that the point on the horizontal axis where the gain of the basket cell population start decreasing is shifted towards lower input magnitudes. This is could be addressed in future studies by using a variable size of basket cells, giving a gradual recruitment of the inhibitory population, and should thus not be considered as problematic as the diverging output relations.

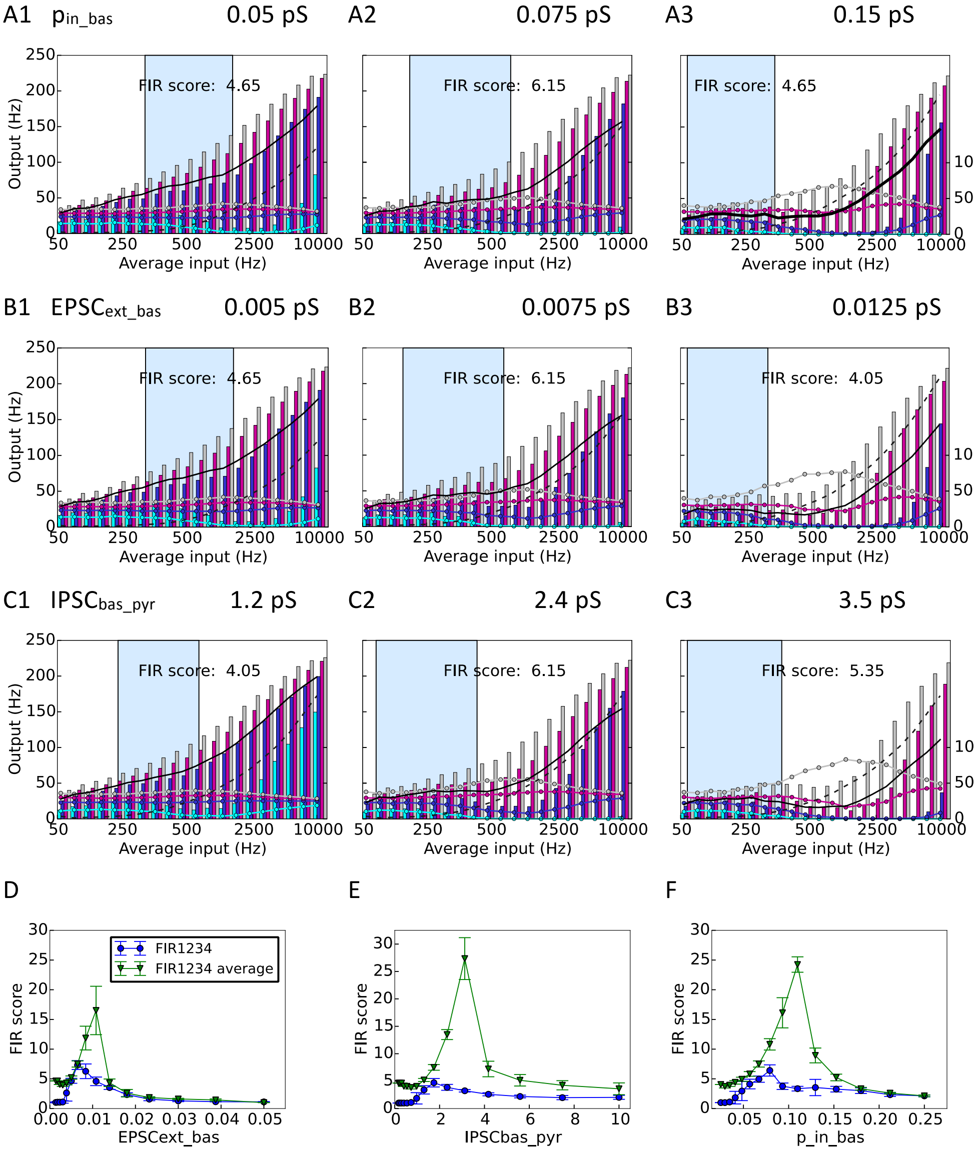

Figure 5.11: The effect of changing the inhibition strength in Model A1. A1-A3) Changing p_{in_bas}. From right to left p_{in_bas}: 0.05, 0.075 and 0.15. B1-B3) Changing $EPSC_{ext\_bas}$. From right to left $EPSC_{ext\_bas}$: 0.005, 0.0075 and 0.0125 pS. C1-C3) Changing $IPSC_{bas\_pyr}$. From right to left $IPSC_{bas\_pyr}$: 1.2, 2.4 and 3.5 pS. D-F) Parameter search over D) $EPSC_{ext\_bas}$ E) p_{in_bas} F) $IPSC_{bas\_pyr}$. Blue line = FIR score. Green line = FIR1234 average score. All scores are averaged over 5 different random seeds. Note how the network is quite sensitive to tuning, and how the peak of the average score is shifted towards a slightly stronger inhibition than the best FIR1234 score.

More on output relations

As for Model A the relations between the outputs for different input relations than those in the FIR tests was investigated. The relationships between the outputs for a hypercolumn with two active minicolumns, as a function of the inputs to the two minicolumns is seen in Figure 5.12 A. It can be seen that output relations are not stable for increased input magnitude. It is also clear that they diverge faster for more dissimilar inputs. Compare with the theoretical normalization model in Figure 5.12 B.

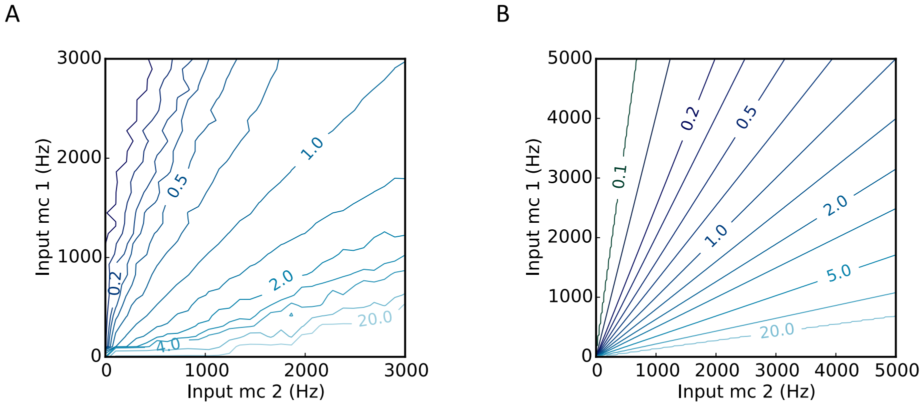

Figure 5.12: Output relations for a hypercolumn with two minicolumns as a function of input to the two minicolumns. A) Model A1 with standard parameters and $\text{EPSC}_{\text{ext\_bas}}$ = 0.0075 pS. If we follow a straight line in the graph that means input relations are kept constant. Note how output relations are not stable across such a line except for low input magnitudes. B) For reference, the same graph for the theoretical normalization model. Note that independently of the relationship between the inputs the relation between the two outputs stays constant for increasing input magnitude.

IO test

The result of the IO test for Model A1 with $\text{EPSC}_{\text{ext\_bas}}$ tuned to the best value found in the previous section (0.0075 pS) is seen in Figure 5.13. Note that the shift in the IO curve is mainly subtractive for higher input magnitudes. However there is also a larger gain change, compared to Model A standard. This is even more pronounced in the low input region, 0 - 500 Hz, and for smaller modulatory inputs. This result is in accordance with what we saw in the FIR test (Figure 5.11 A2); between 50 and 500 Hz is the network approximately normalizing, but for higher input magnitudes there is a switch to WTA dynamics. For this model versions, in the same way as for Model A standard, the region of approximately constant average is considerably longer than the region of stable output relations. It is thus primarily the latter that limits the normalizing ability. Because of this the reasons for why output relations are diverging and if there are mechanisms counteracting this, are studied in the next section.

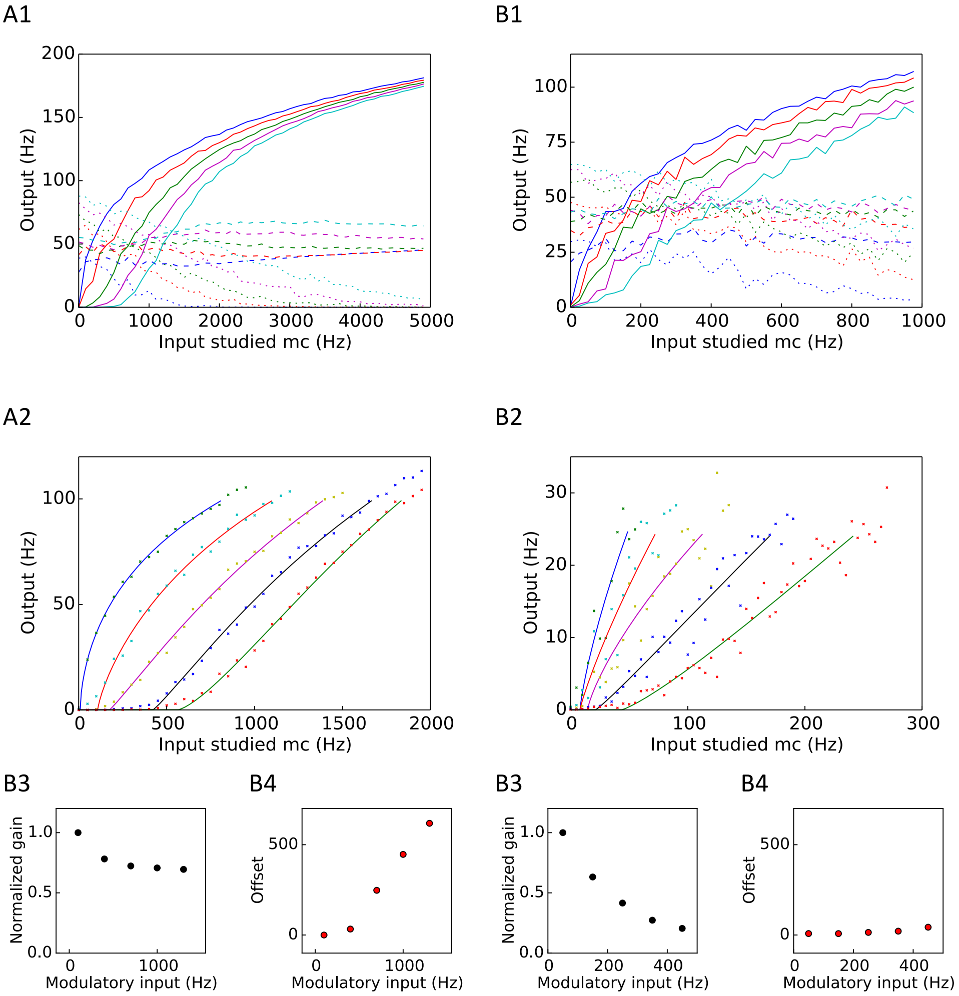

Figure 5.13. IO test for feed-forward inhibition only. Standard parameters from Table 4.2 except EPSC$_{ext\_bas}$ which is set to 0.0075 pS. A1) IO test. Modulatory inputs: 100, 400, 700, 1000 and 1300 Hz. A2) Hill function fits to data points in A1 below 100 Hz. A3) Normalized gain for curves in A2 between 5 and 100 Hz. A4) Offset for curves in A2. B1) IO test. Modulatory inputs: 50, 150, 250, 350 and 450 Hz. B2) Hill function fits to data points in B1 below 20 Hz. B3) Normalized gain between 2.5 and 25 Hz for curves in B2. B4) Offset for curves in B2. Note that for the low input region and smaller modulatory inputs there is a considerable gain change.

5.2.2 Feed-forward inhibition vs output relations

To analyze the behavior of Model A1 a simple mathematical model of the network was constructed. One building block was that the response functions of the minicolumns and the basket cell populations are sigmoidal. A second one was that inhibition have been shown to primarily have a subtractive modulatory effect on those population response curves.

An expression for the inhibition each minicolumn receives - and note that each minicolumn will receive *the same amount* of inhibition - would thus be:

$$h = IPSC_{bas\_pyr} N_I \, R^b_{max} g(p_{in\_bas} EPSC_{ext\_bas} \sum_j x_j - v) \tag{5.4}$$

$$\rightarrow h = kg(d\sum_j x_j - v)$$

$$g(x) = \frac{x^{n_b}}{\sigma_b^{n_b} + x^{n_b}}$$

Where k represent the strength of the inhibition from the basket cells to the pyramidal cells, d the strength of the feed-forward drive to the basket cells and v the position of the basket populations rheobase in relation to the minicolumns. Subscript b in function g means basket cell parameters. The assumption that adding inhibition to the minicolumns, has the same effect as subtracting a constant from the independent variable of the response function was made. This means f(x, h) = f(x - h), where x is excitatory and h inhibitory drive. The output from minicolumn i can then be approximated as:

$$y_i = f(x_i - kg\big(d\textstyle\sum_j x_j - v\big)) \qquad (5.5)$$

For convenience the coordinate system is shifted to the right in relation to total input magnitude a, where an input vector as previously is described as a·[c1, c2, .. cn] where c_i represent the percentage of total input magnitude minicolumn i receives. This gives:

$$f(m_i + ac_i - kg\big(d\textstyle\sum_j ac_j\big)) \qquad (5.6)$$

$$f(x) = R_{max}\,\frac{x^{n_p}}{\sigma_p^{n_p} + x^{n_p}}$$

Where f(x) is the IO curve of the pyramidal population also approximated as a sigmoidal function. If this model is studied it explains why input relations tend to diverge, and why this happens faster for a stronger inhibitory drive. To get a constant average the response of the basket cells need to be approximately linear for increased input magnitude (here increased a). As discussed before it is not really but for a short range it can be approximately linear. For the range where it is we have:

$$\begin{aligned} y_i &= f(m_i + ac_i - kd\textstyle\sum_j ac_j) \\ &= f(m_i + a\big(c_i - kd\textstyle\sum_j c_j\big)) \qquad (5.7) \\ &= f(m_i + aC_i) \end{aligned}$$

Thus if $C_i > 0$ the output from that unit will grow and if $C_i < 0$ it will diminish. To have a constant average the inhibition need to cancel out the excitation when the input magnitude is increased. This means for any minicolumn in the hypercolumn Equation 5.10 must hold:

$$C_i = 0 \rightarrow kd = \frac{\sum_j c_j}{c_i} \qquad (5.8)$$

If all minicolumns have the same input (that is the input vector is a*[1, 1, 1, 1] this is value of k · d could be chosen and the network would keep average output stable (see e. g. Figure 4.5). The problem is when we have different inputs to the minicolumns. For a network with two minicolumns, input $ac_1 < ac_2$, and still approximately linear inhibition the average is directly proportional to:

$$y_1 + y_2 = f(m_1 + aC_1) + f(m_2 + aC_2) \tag{5.9}$$

To keep this constant either both outputs have to be constant (which will only be the case if $C_1 = C_2 = 0$) or one output has to grow with the same rate as the other one decrease. The first case is not possible since $c_1 < c_2$ which implies $C_1 < C_2$. For the second approach if making the assumption that the pyramidal response curve is also approximately linear, the appropriate value for k·d could be calculated:

$$\begin{gathered} y_1 + y_2 = r(m_1 + aC_1) + r(m_1 + aC_1) \qquad (5.11) \\ = 2r(m1 + m2 + a(c_1 + c_2 - 2kd\textstyle\sum_j c_j)) \\ => \; kd = \frac{c_1 + c_2}{2\sum_j c_j} \end{gathered}$$

This also implies $C_1 = - C_2$. What it means is that if we cannot choose the inhibition to cancel out the excitation for all minicolumns (which we cannot since they receive different proportion of excitation and the same amount of inhibition) to keep the average constant *one output has to grow at the same rate as the other decrease, and consequently the difference between them will increase*. This was exactly what was seen in the FIR graphs from Model A standard and A1. Although we here assumed a linear response curve for the pyramidal cell population (primarily to simplify the calculation), the specific *shape* of the pyramidal response curve in fact does not matter as long as it is a strictly increasing function. The same principle would thus apply if it resembles a power law, exponential or logarithmic function that is for all parts of its sigmoidal response curve.

The shape of the IO curve will however affect how *fast* the output relations will grow apart, since the derivative of f (c_1a) / f (c_2a) with respect to a is dependent on the derivative of the pyramidal population response curve f, as well as the derivative of the basket population response curve g. That allowing a certain increase in input magnitude will counteract diverging output relations can also be understood if the decreasing gain for the second part of a sigmoidal response curve is considered. A minicolumn with a larger c_i value will still get a larger increase in excitation for a certain increase in average input magnitude. However this will be partly counteracted by that gain is lower for a minicolumn that already receives a large input because of the decreasing Hill function derivative. This explains why perfect average control tend to counteract stable input relations.

However it is important to remember that this model will only be a good representation of system functionality when a single minicolumn response curve is indeed affected subtractively by increased inhibition.

5.2.3 Tuning EPSC$_{ext\_bas}$ and IPSC$_{bas\_pyr}$ together

Even if average control is not sufficient to create a normalizing network, it is a requirement. Since the diminishing derivative of the inhibitive response curve was shown to be a major factor limiting the range over which average can be kept stable, it is interesting to see if there is some way this could be counteracted. It was seen in Section 5.2.1 that decreasing the input to the basket cells do scale their output curve on the horizontal axis (equivalent to dividing the independent variable of a function by a constant), and that increasing IPSC$_{bas\_pyr}$ increase the resulting amount of inhibition from a specific firing rate in the basket cell population. So the idea tested in this section is what would happen if $EPSC_{ext\_bas}$ is decreased while at the same time as $IPSC_{bas\_pyr}$ is increased. This should give the basket cell response curve a longer region of approximate linear growth as illustrated in Figure 5.14 and thus make it possible to keep optimal inhibition strength over a larger range of input magnitudes.

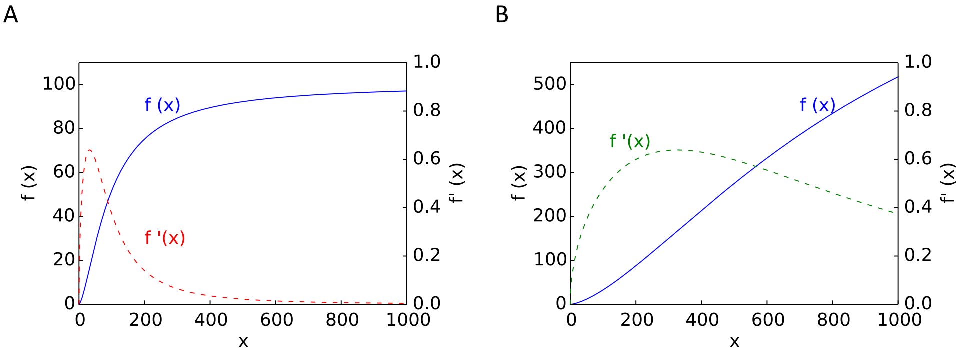

Figure 5.14: The Hill function (Equation 4.4) and its derivative. Continuous line: function. Dashed line: derivative. A) R_{max} = 300, σ = 1500, n = 1.5. B) R_{max} = 3000 and x = x/10. Note that there is a much slower change in the derivative in B which is in analogue with how decreasing $EPSC_{ext\_bas}$ and increasing $IPSC_{bas\_pyr}$ was hypothesized to affect the derivative of the basket population response curve in the network model.

For this purpose $IPSC_{bas\_pyr}$ was increased to - 25.0 pS (this gives an EPSP of 6.1 mV) and other neuronal parameters was tuned for optimal performance. As illustrated in Figure 5.15 the approach indeed works as a way of extending the region of stable average. However more interesting is that it also extends the range of both input magnitudes and input relations over which the network is approximately normalizing. The divisive scaling of the IO curve also for higher input magnitudes seen in Figure 5.15 A1, is a clear qualitative difference from previous models. Here a pronounced gain change is present also for larger modulatory inputs. This together with the absence of subtractive shift give that the modulatory effect of the network is now more accurately described by the normalization model than the subtractive inhibition model.

However as seen in Figure 5.15 B the network is not normalizing over the whole input range and Figure 5.15 C show that the normalization works better for inputs of similar magnitude, as for the previous model versions. However, what is troubling about this model version is that the basket cells now fire at a very low rate (below 20 Hz) and also lower than the pyramidal cells. In addition these IPSP values could not be considered biologically realistic if compared with data from (Thomson et al. 2002). It should be however be remembered that if we only wish to keep *the average* constant, this mechanisms does not specifically require a low firing rate but that the basket cell do have a *stable derivative* over quite a large range of inputs. This might also be achieved by increase the number of basket cells or by using a group of different sized basket cells, where the smaller start up for lower input rates and the larger ones gets recruited for a larger input magnitude (Lansner 2014). However as discussed previously keeping *average* constant does not necessarily give a normalizing network.

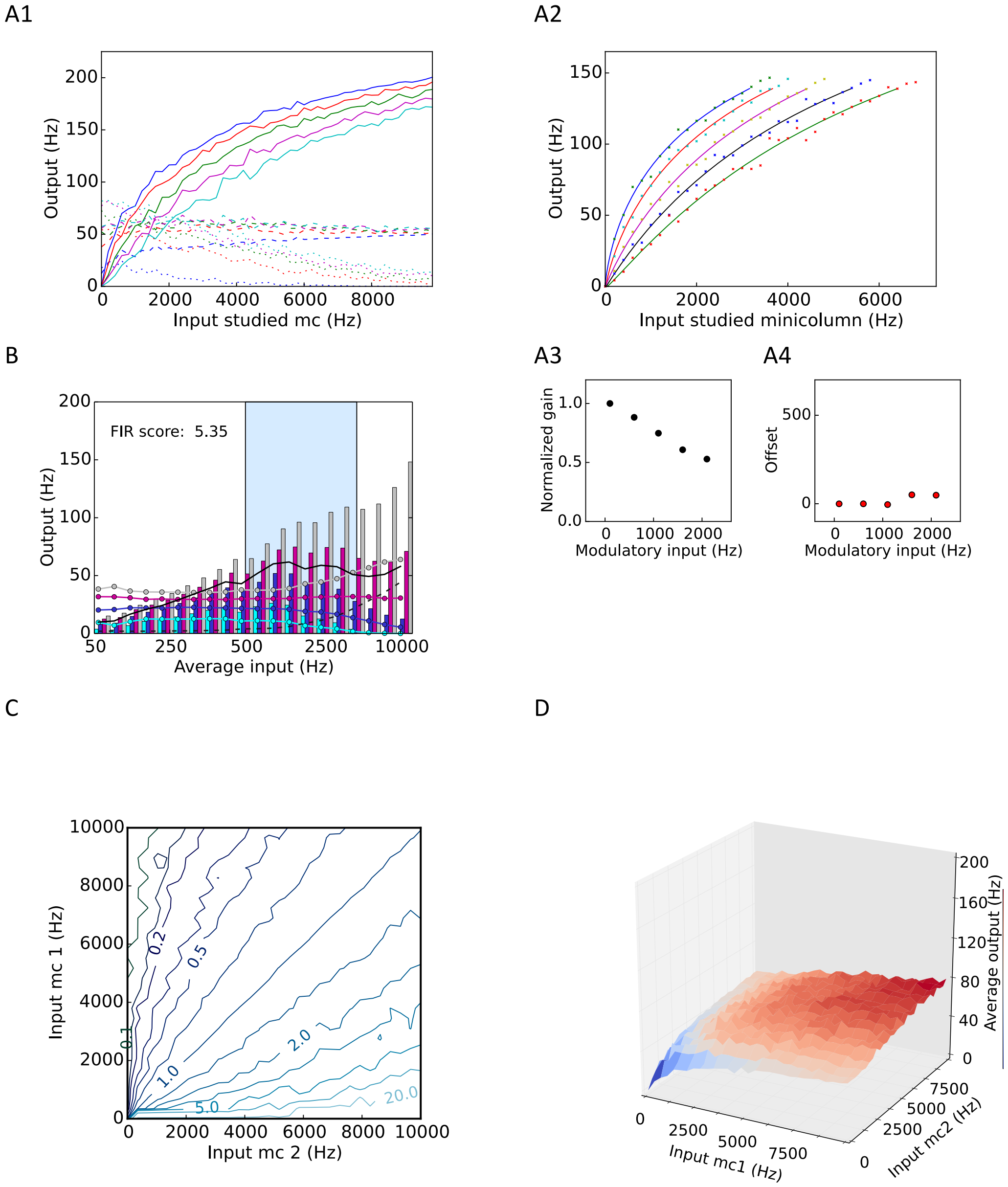

Figure 5.15: The effect of increasing IPSC$_{bas\_pyr}$ and decreasing EPSC$_{ext\_bas}$ while keeping inhibition strength constant in Model A1. Differences from parameters in Table 4.2: IPSC$_{bas\_pyr}$ = 25.0 pS, EPSC$_{ext\_bas}$ = 0.0014 pS, noise$_{bas}$ = 5900 Hz, p$_{in\_bas}$ = 0.05. A1) IO test. Modulatory inputs: 100, 400, 700, 1000 and 1300 Hz. A2) Hill function fit (Equation 4.4) to the data points in A1. A3) Normalized gain for curves in A2. A4) Offset for curves in A2. B) FIR1234 test. The reason there is enough inhibition from a very low activity in the basket cell population is because of the large IPSC$_{bas\_pyr}$. C) Relationship between outputs for a network with two active minicolumns, as a function of input to the respective minicolumns. D) As C but average activity in the network as a function of input to the two minicolumns.

This was also confirmed by instead of using a large IPSP using a larger number of basket cells in the inhibitory population. The result seen in Figure 5.16 show that this does not give same results, but instead this network is as the previous network model versions closer to the subtractive inhibition model. So the results of this experiments are maybe most important in the aspect that they point to a mechanism that can indeed create a more normalizing network. How this mechanism works was investigated further, and the results are presented in the next section.

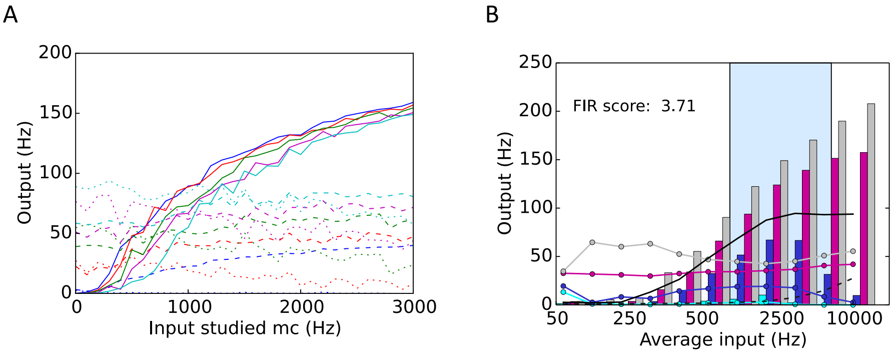

Figure 5.16: FFW network with larger number of basket cells, N_I = 160. A) IO test. Modulatory input: 100, 400, 700, 1000 and 1300 Hz. Note that the gain change seen when using a large IPSP is not seen when instead using a larger number of inhibitory neurons. B) FIR1234 test. This is very similar to what was seen in Figure 4.5 C for the theoretical subtractive inhibition model.

5.2.4 Noise, gain change and normalization

If the derived mathematical model can explain why output relations are not kept stable for higher input magnitudes, what explains why they are stable for lower input regions? And why does it work better when we decrease $EPSC_{ext\_bas}$ and increase $IPSC_{bas\_pyr}$? An assumption made in the mathematical model was that the IO curve would keep its shape for increased amounts of inhibition. As results from previous model versions using standard parameters show this is a valid approximation for output rates above 10-20 Hz. However the output rates where there is gain change for Model A and A1 with standard parameter values are lower than that. And what effect the large $IPSC_{bas\_pyr}$ might have on the model neuron IO curve is not yet investigated.

To gain a wider perspective an experiment to study how the response curve is affected by increased inhibition for standard vs large IPSP sizes was conducted. The setup was based on the fact that for a specific network setup in Model A1 the activity in a minicolumn will only be affected by two state variables: i) The amount of excitatory input to the pyramidal cells and ii) The amount of excitatory input to the basket cells (since the input is approximated by a Poisson process a specific input frequency is tied to a specific variance). Thus if the output firing rate of a single minicolumn is plotted as a function of these two variables it is possible to study how the IO curve change for increased amounts of drive to the basket cells as well as how a minicolumn will “travel” in this state space when input magnitude is increased.

The results are seen in Figure 5.17. Since for the feed-forward setup a certain amount of inhibition is always coupled with a certain amount of excitation, each minicolumn would for increased input magnitude travel along a straight line in this graph/ state space. The graphs visualize what was shown mathematically in the Section 5.2.2 about why input relations change for increased input magnitude: To keep the average constant it is necessary to choose the inhibition strength (slope of the line the minicolumn will follow in this graph) so that the average output follow one of the contour lines in the plot. But if minicolumns have different c_i values, they will have a steeper or lower slope. If there is no widening of the contour lines (as would be the case if the response curve is affected divisively) the difference between them will be magnified with growing input magnitude.

These results also visualize why it is harder to keep the relations between the outputs for more dissimilar inputs: The difference between the paths for the different minicolumns is then larger. Comparing Figure 5.17 A and B with Figure 5.17 C and D it is also clear that the approximation of a primarily subtractively shifted curve is valid when the standard parameters are used but not for the “large IPSP network”. It can also be seen that for a larger IPSP the variance in the output is larger, reflecting a larger neuronal membrane variance. As discussed in Section 3.4.3 increased variance in the input gives a more gradual onset of firing. This is seen as larger spacing between contours for higher amounts of inhibition, which partly counteract the increased differences between the outputs. Thus increased neuronal membrane variance seem to be the primary reasons we get a better approximate normalization for a network when a larger IPSP is used.

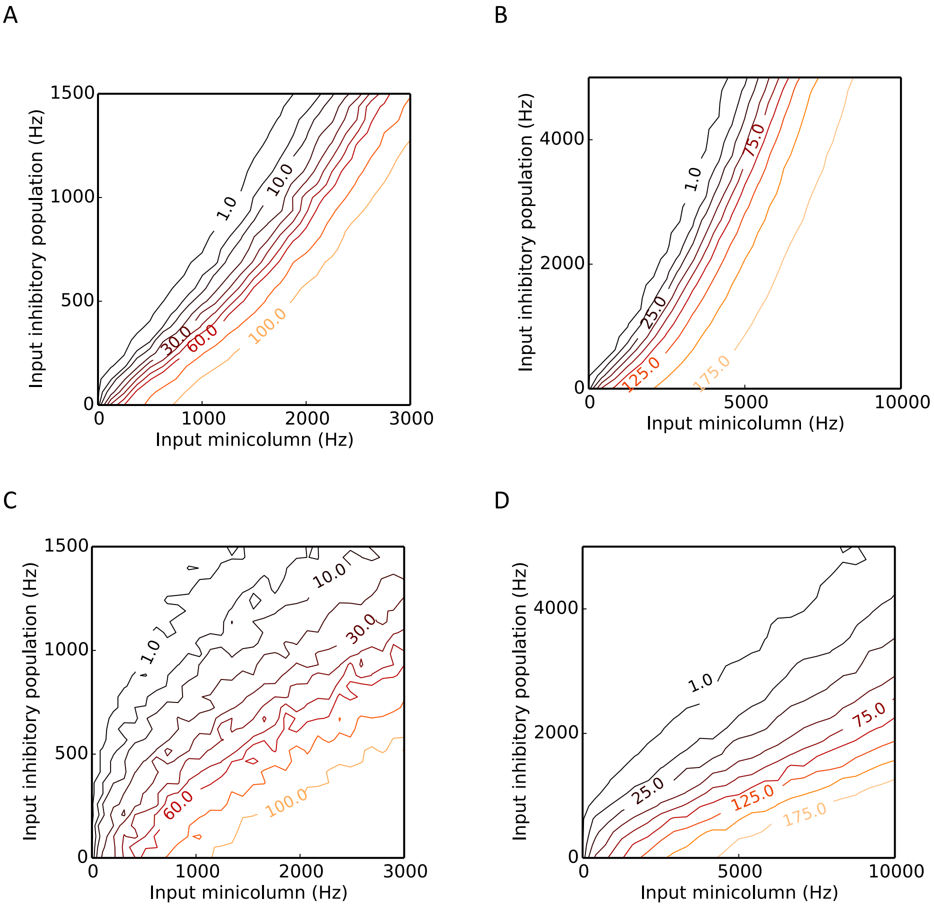

Figure 5.17: Output from single minicolumn in Model A1 standard parameters for different amounts of excitation and inhibition. A, B) Model A1 standard parameters. Note for that basket cells output get saturated for higher input. For the low input region it is possible to choose a path (straight line) that keep average constant but as input increase this is no longer possible. C, D) IPSP = 40 pS, EPSP$_{ext\_bas}$ = 0.0014 pS. This parameter setting both smoothen the onset of firing and keep basket cells from saturating. D) Also for larger input magnitudes output relations can be kept constant for more dissimilar inputs.

Since it is interesting to know if a large IPSP change also the response curve of a single neuron this was as well investigated. The pyramidal cell response curve was plotted for increased amounts of inhibition for three different IPSP sizes: 2.6, 12.0 and 25.0 pS. The rate of the inhibition was adapted to give approximately the same inhibitory drive for the three IPSP sizes. The results seen in Figure 5.18 show that there is a clear difference in the modulatory effect of an inhibitory Poisson input depending on the IPSP size also for a single neuron: The larger IPSP the better can the effect be described as divisive instead of subtractive. This indicate that the modulatory effect seen in the network for a large IPSP is most likely a superposition of modulatory effects implemented by individual neurons.

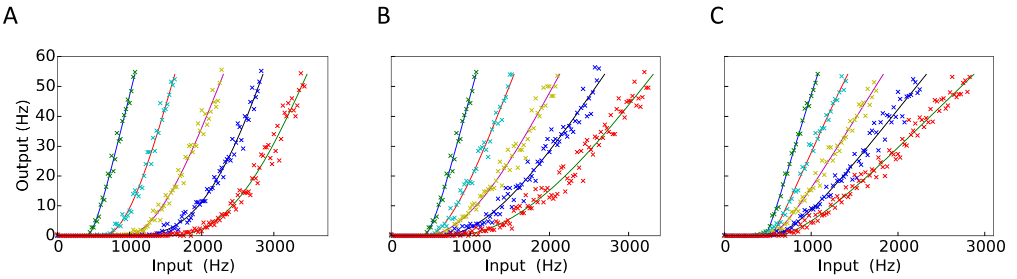

Figure 5:18: How increased IPSP size change the modulatory effect of Poisson inhibition on a single model neuron. Lines are Hill function fits (Equation 4.4) using data points below 50 Hz. A) IPSP = 2.6 pS. Inhibition: 0, 200, 400, 600 and 800 Hz. B) IPSP = 12.0 pS. Inhibition: 0, 50, 100, 150 and 200 Hz. C) IPSP = 25.0 pS. Inhibition: 0, 25, 50, 75 and 100 Hz. Note that for a larger IPSP the modulatory effect of from an inhibitory input is transformed from subtractive to divisive.

Table 5.2: FIR test results for different versions of Model A1. Numbers are FIR scores averaged over 5 different random seeds with standard deviation. Green: best scores.

	Model A1	**Model A1**	**Model A1, large IPSP**
	standard	**EPSC$_{ext\_bas}$**	**IPSC$_{bas\_pyr}$ = 25.0 pS**
	parameters	**= 0.0075 pS**	**EPSC$_{ext\_bas}$ = 0.0014 pS**
FIR1234	5.15 ± 0.95	**5.86 ± 0.82**	5.86 ± 0.69
FIR1234 average	13.60 ± 1.90	10.0 ± 1.45	**16.92 ± 1.73**
FIR1200	3.62 ± 0.21	3.24 ± 0.33	**6.73 ± 0.79**
FIR1200 average	4.04 ± 0.0	4.16 ± 0.24	**6.73 ± 0.79**

Concluding remarks Model A1

So in conclusion the feed-forward network can using standard parameters approximate normalization over a larger range than the subtractive inhibition or output gain model. Between 50 and 750 Hz, input magnitude can be increased by about 5 times while keeping output relations approximately stable. However it is also for this range of input magnitudes only a good approximations if the inputs are quite similar in magnitude. Also for the tuned network with very large IPSP this effect is present even if it can handle slightly larger input magnitudes and differences between inputs. The FIR scores for different versions of Model A1 are seen in Table 5.2.

It was also seen that inhibition strength can be tuned to give an inhibition that over a certain range of input magnitudes is approximately balanced to the excitation. However there will always be a point where the inhibition to the minicolumns grows slower than their excitatory input. It was also shown that a constant average does not necessarily give a normalizing network. The most interesting result from using a very large IPSP size is that it shows a mechanism that can be used to create a normalizing network - *an increase in neuronal membrane variance coupled to an increase in modulatory input.* Even if the way this was done could be considered unbiological, there might other ways to get a similar effect.

So could feed-back inhibition give a different effect and overcome some of the problems we face when trying to create a normalizing network with only feed-forward inhibition?

5.3 Model A2 - Feed-back inhibition

This section study Model A2 with only feed-back inhibition, to see what part the feed-back inhibitory circuit plays for network functionality. A difference in network setup is that for Model A2 there is in the standard setup no noise to the basket cell population. The feed-back mechanism makes the amount of noise to basket cells less important since the basket cell population is driven directly by the activity in the minicolumns. The network characteristics for a single input vector $\overline{x}$ = [400, 800, 1200, 1600] Hz is seen in Figure 5.19. Those are similar to what was seen for Model A (Figure 5.5), with the strong oscillatory activity as the main difference from Model A1.

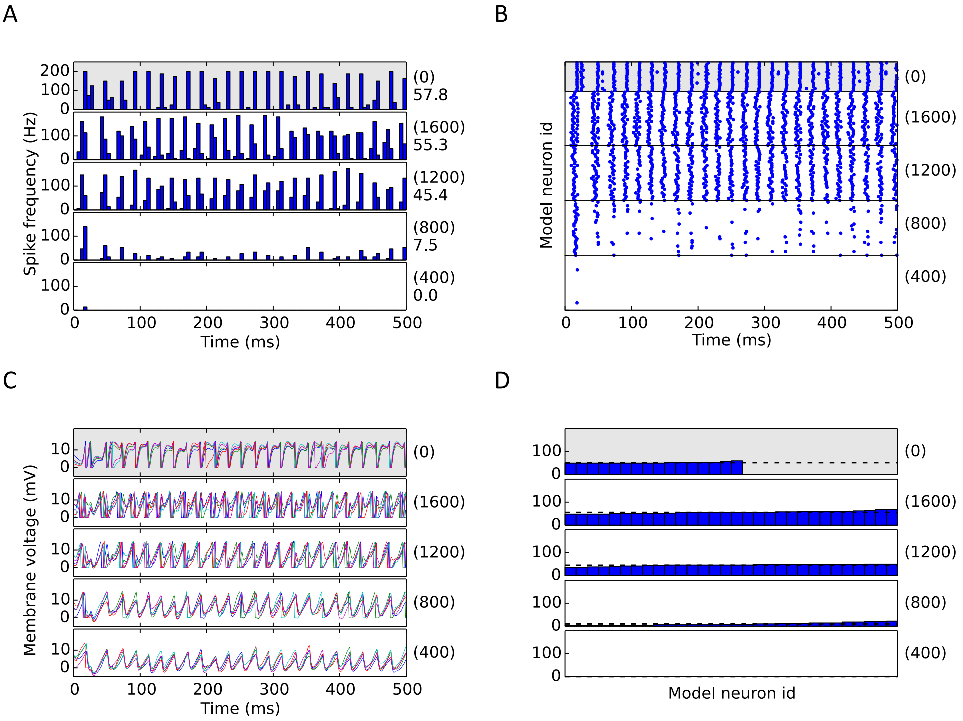

Figure 5.19: Network activity for a single input vector with $\overline{x}$ = [250, 500, 1000, 2000] Hz input. A) Spike histogram. B) Raster plot. C) Voltage traces for 5 randomly chosen neurons in each minicolumn. D) Distribution of spikes for neurons in respective minicolumn.

5.3.1 Tuning inhibition strength

Since there is only feed-back inhibition in Model A2 the two parameters tuned to vary inhibition strength was $\text{IPSC}_{\text{bas\_pyr}}$ and $\text{EPSC}_{\text{pyr\_bas}}$. The same tuning tests done for Model A1 were performed also for this model where the average of five

FIR1234 scores using different random seeds was used as a measure of the networks normalizing ability. The objective was also to see if there were any qualitatively different effects from previous model versions especially regarding:

- Average control in the network.
- Relationships between outputs for increased input magnitude.

The result of tuning $\text{EPSC}_{\text{pyr\_bas}}$ is seen in Figure 5.20 A1-A3 and the graphs are quite similar to those seen for the feed-forward model. The same characteristic “bubble” is seen, where input relations first diverge and later converge. It is also seen that for a larger $\text{EPSC}_{\text{pyr\_bas}}$ the basket cells reach higher firing rates for smaller input magnitudes. A difference is though that we see the best score on the FIR1234 fest for a very small value of $\text{EPSC}_{\text{pyr\_bas}}$. This is since at the onset of the inhibition there is a break in the otherwise constantly growing average. The result of instead tuning $\text{IPSC}_{\text{bas\_pyr}}$ is seen in Figure 5.20 B1-B3.

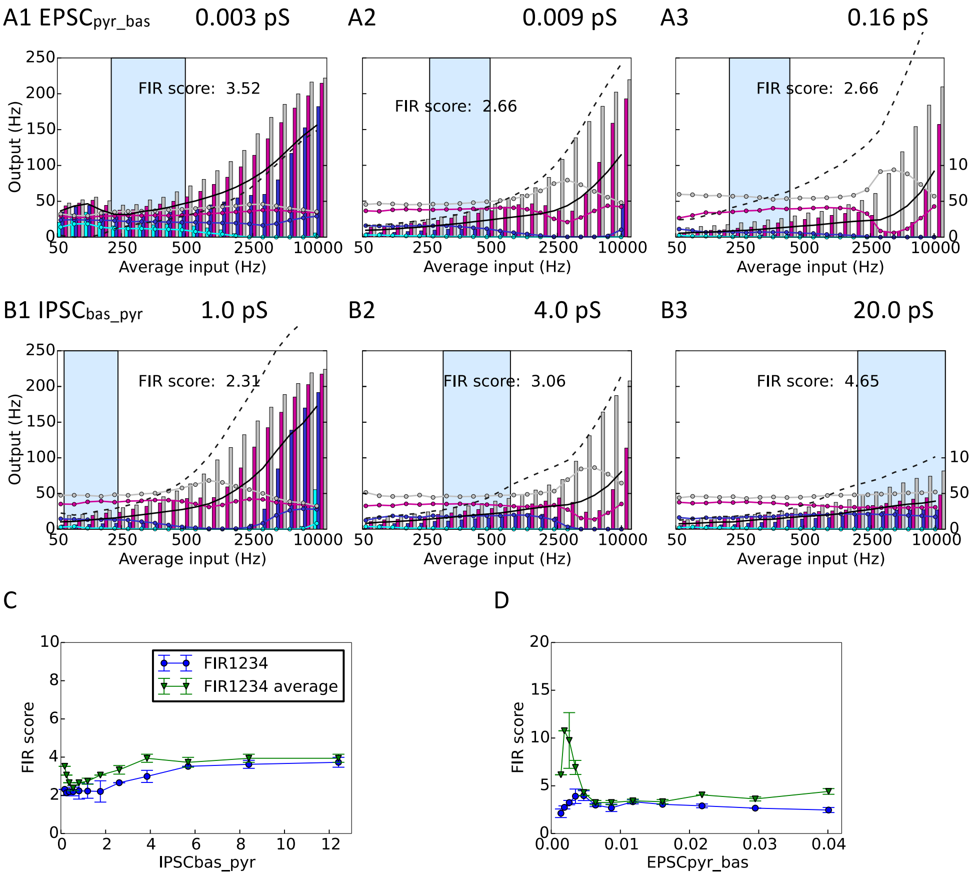

Figure 5.20: The effect of tuning inhibition strength in Model A2. A1-A3) Tuning $\text{EPSC}_{\text{pyr\_bas}}$. From right to left $\text{EPSC}_{\text{pyr\_bas}}$ = 0.003, 0.009 and 0.16 pS. B1-B3) Tuning $\text{IPSP}_{\text{bas\_pyr}}$. From right to left $\text{IPSC}_{\text{bas\_pyr}}$ = 1.0, 4.0 and 20.0. pS. C) Parameter search over $\text{IPSC}_{\text{bas\_pyr}}$ D) Parameter search over $\text{EPSC}_{\text{pyr\_bas}}$. Green curve = FIR1234 average score. Blue curve = FIR1234 score. All scores are averaged over 5 different random seeds. Note in D that average control gets considerably worse for increased drive to the basket cell population because of saturation. However as seen in C the network is quite insensitive to changes in $\text{IPSC}_{\text{bas\_pyr}}$.

The histograms for smaller values of IPSC$_{bas\_pyr}$ are quite similar to Model A1, but in contrast to Model A1 the network has the best performance for very large IPSP values. Another characteristic effect is that also for very large IPSP values, there is, in contrast to Model A1, no decrease in average for large input magnitudes.

As seen in the parameter search plots (Figure 5.20 C and D) the feed-back network is indeed less sensitive to a change in parameter values. This is because of the feed-back loop: If we e. g. increase IPSC$_{bas\_pyr}$ this gives more suppression for the same basket population firing rate; but then the minicolumns firing rate will decrease, which will in turn decrease the drive to the basket cells. This until a new stable state is reached with slightly lower firing rate for both excitatory and inhibitory populations. For decreasing IPSC$_{bas\_pyr}$ the stable state will instead be found for a slightly higher firing rate in the pyramidal population. There is thus a dynamic balance between the firing rate in the pyramidal and basket cell populations not present in the feed-forward network. However the range over which the network pass the FIR1234 test is slightly smaller than for Model A1. This is mainly because it is limited by the strictly growing average.

IO test

The results from the IO test for Model A2 with EPSC$_{pyr\_bas}$ set to 0.003 pS, are shown in Figure 5.21. The IO curves show a clear subtractive shift for higher output rates and modulatory inputs. There is as for Model A1 also a small gain change present, more pronounced for lower output rates and modulatory inputs. It can be seen in Figure 5.21 C that output relations are not stable for increasing input magnitude and in Figure 5.21 D that average output vary both with increased input magnitude and with relationships between inputs.

5.3.2 Keeping average constant with feed-back inhibition

A problem for the feed-back only network is that it is unable to implement as good average control as for feed-forward inhibition. The range over which Model A2 can keep average approximately constant, is a little less than half that of Model A1. Increasing inhibition strength does not seem to improve average control very much. To understand why a mathematical model of Model A2 in analogue with the one used for the feed-forward network was constructed. The result is seen as Equation 5.13 that describes how the output from minicolumn i depends on input, inhibition strength and activity in other minicolumns. Also here f (x, h) = f(x - h) was assumed where x is excitatory and h inhibitory drive.

$$y_i = f(m + x_i - kg(d \textstyle\sum_j y_j)) \qquad (5.13)$$

Here k correspond to strength of the drive from basket population to the minicolumn, and d strength of the drive from the minicolumn to the basket cells population. This model explain why average control is a problem for a network using only feed-forward inhibition. Consider the case when all inputs are equal and assume that average could be kept constant for increased input magnitude. This would imply that the inhibition cancels out the increased excitation. But inhibition is a strictly growing function of the outputs. This means to get larger inhibition larger outputs are needed, thus creating a paradox and disproving the original assumption. A similar argument holds when the inputs are not similar since both minicolumns would receive increased excitation – and the inhibition thus must increase to keep average stable.

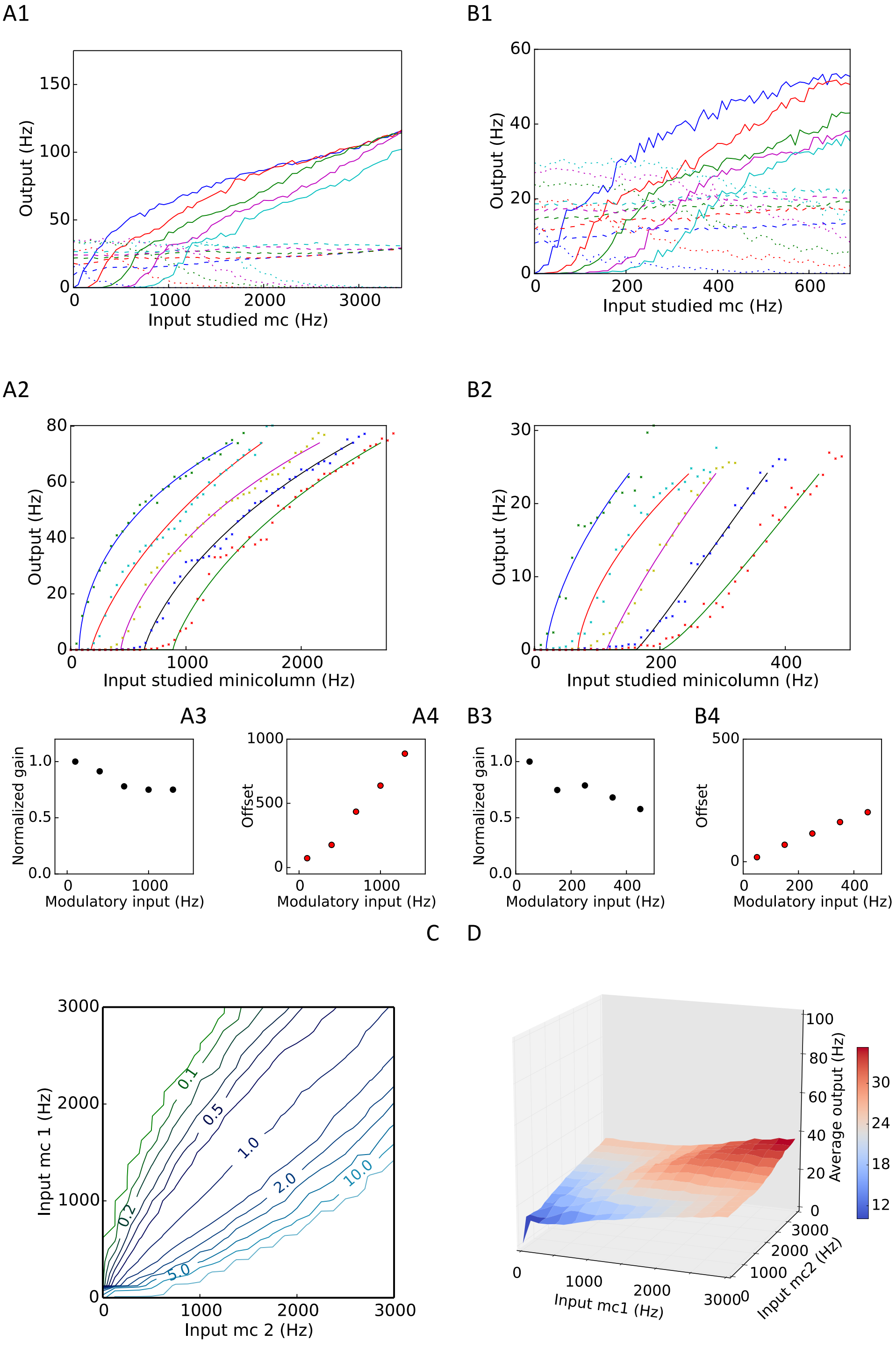

Figure 5.21: IO test for Model A2. Standard parameters except EPSC$_{pyr\_bas}$ = 0.003 pS. A1) IO curve. Modulatory inputs 100, 400, 700, 1000 and 1300 Hz. A2) Hill function fit (Equation 4.4) to data points in A1 below 80 Hz. A3) Normalized gain for fitted curves in A2. A4) Offset for fitted curves in A2. B1) Io curve, modulatory input 50, 150, 250, 350 and 450 Hz. B2) Sigmoid fit to data points in B1 below 20 Hz. B3) Normalized gain for curves in B2 between 2.5 and 20 Hz. B3) Subtractive offset for curves in B2. C) Relationship between outputs for a network with two active minicolumns, as a function of input to the respective minicolumns. D) As C but average activity in the network.

However as for the feed-forward model the assumption that increased inhibition only shift the curve to the right on the horizontal axis is a simplification, and there are conditions when it is not valid such as when the inhibition starts up (see e. g. Figure 5.20 A1). But this is only the case just when the inhibition starts up and after that the difference in shape between response curves for increased modulatory input is for standard parameters not large enough to allow good average control. This means though an increased inhibitive strength will give a slower growth in average, no choice of EPSC$_{pyr\_bas}$ or IPSC$_{bas\_pyr}$ will in contrast to the feed-forward network keep it constant.

Relationships between outputs

Considering the relationships between the outputs for an input vector $\bar{x}$=a·[c$_1$, c$_{2,}$ .. c$_n$] the mathematical model described by Equation 5.13 gives:

$$\frac{y_i}{y_j} = \frac{f(m_i + \ ac_i - kg(d\sum_j y_j))}{f(m_j + \ ac_j - kg(d\sum_j y_j))} \qquad (5.14)$$

As in the feed-forward network, we see that both minicolumns receive the same amount of inhibition but different amounts of excitation with increased input magnitude (there is nothing in the feed-back circuit that change this). This means the argument from the model of the network with only a feed-forward inhibitory circuit still holds: A minicolumn with a smaller c$_i$ will for increased input magnitude either decrease or grow slower than a minicolumn with a larger c$_i$ thus magnifying the difference between them. The increasing average of this network will as for the feed-forward model partly counteract this. When considering the results of the IO and FIR tests this simple model is a good description of the functionality of Model A2 when slightly tuned standard parameters are used.

5.3.3 Tuning EPSC$_{pyr\_bas}$ and IPSC$_{bas\_pyr}$ together

Also for Model A2 the same approach of increasing IPSC$_{bas\_pyr}$ and decreasing EPSC$_{pyr\_bas}$ was tried. This did increase the region of approximately stable average as well as the range of input magnitudes over which the network could keep output relations approximately stable. This can be seen in Table 5.3 as a considerably better result on the FIR tests. Similar qualitative characteristics for the IO test and sigmoidal Hill function fits, as well as the FIR1234 and FIR1200 test were also seen for this model version. However graphs are not shown but instead similar plots for a large IPSP can be seen in next section for the tuned versions of Model A.

Table 5.3: FIR test results for different versions of Model A2. All scores are averaged over 5 different random seeds. Green - best scores.

	Model A2, standard parameters	**Model A2, tuned** **EPSC$_{pyr\_bas}$ = 0.003 pS**	**Model A2, large IPSP** **IPSC$_{bas\_pyr}$ = 40.0 pS** **EPSC$_{pyr\_bas}$ = 0.0014 pS**
FIR1234	2.74 ± 0.16	3.45 ± 0.55	**6.58 ± 1.80**
FIR1234 average	3.52 ± 1.99	7.09 ± 2.99	**9.36 ± 1.12**
FIR1200	3.72 ± 0.26	2.59 ± 0.14	**7.10 ± 0.63**
FIR1200 average	7.10 ± 2.99	8.14 ± 0.0	**12. 80 ± 1.31**

Concluding remarks Model A2

In conclusion the effects of feed-back inhibition considering normalization ability are similar to those of feed-forward inhibition. The network can approximate normalization over a certain range of input magnitudes and relations, but it is not possible to extend this region very much by only tuning inhibition strength. For the tuned model version the FIR scores are slightly worse than for Model A1. Since the feed-back or feed-forward inhibitory circuit seem to complement each other, in that the feed-forward circuit is better suited to keep average stable, while the feed-back circuit dynamically adapts inhibition and thus makes the network a less sensitive to small changes in parameter values, the following investigations were performed on Model A, which includes both these circuits.

5.4 Model A revisited

This section continue the study of Model A, that is the network version including both the feed-back and the feed-forward inhibitory circuit. First the effect of a larger IPSC$_{bas\_pyr}$ was investigated, and then several changes to the model was done to see if there might as well be other mechanisms that could create a normalizing circuit.

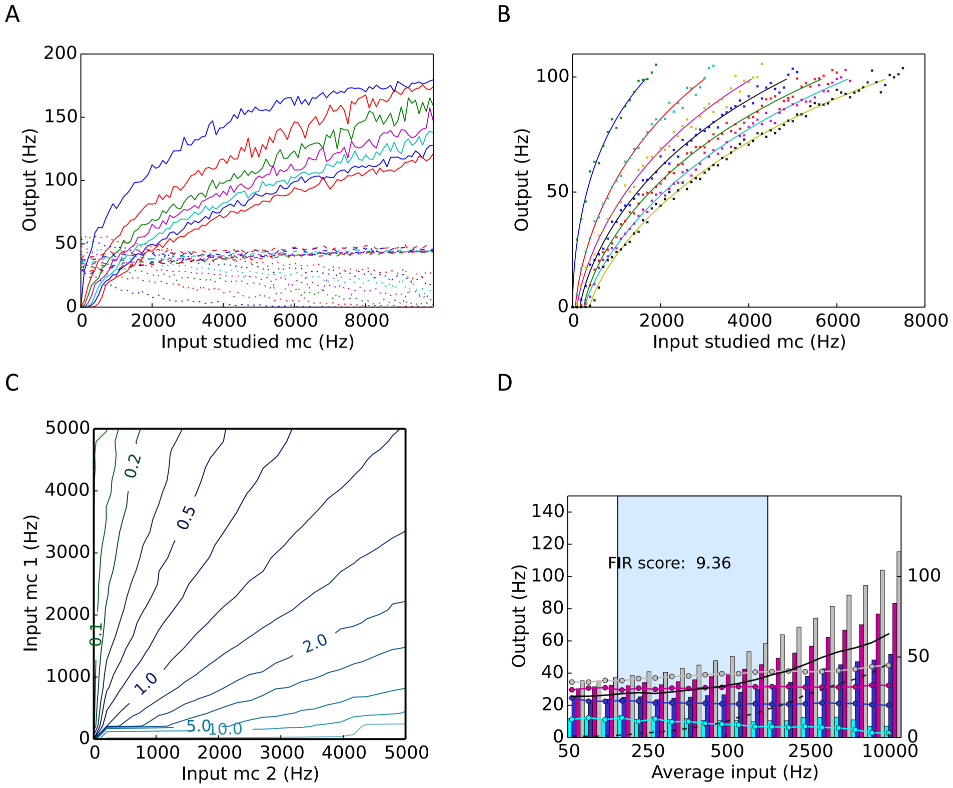

Figure 5.22: Model A large IPSP. Parameters changed from those seen in Table 4.2: IPSC$_{bas\_pyr}$ = 40.0 pS, EPSC$_{pyr\_bas}$ = 0.001 pS, EPSC$_{ext\_bas}$ = 0.001 pS, p$_{in\_bas}$ = 0.05 and noise input bas = 5000 Hz. A) IO test, input to the other minicolumns 100, 400, 700, 1000 and 1300 Hz. B) Hill function fit (Equation 4.4) to data points in A. C) Relationships between outputs for two active minicolumns, as a function of the input to those minicolumns. D) FIR1234 test.

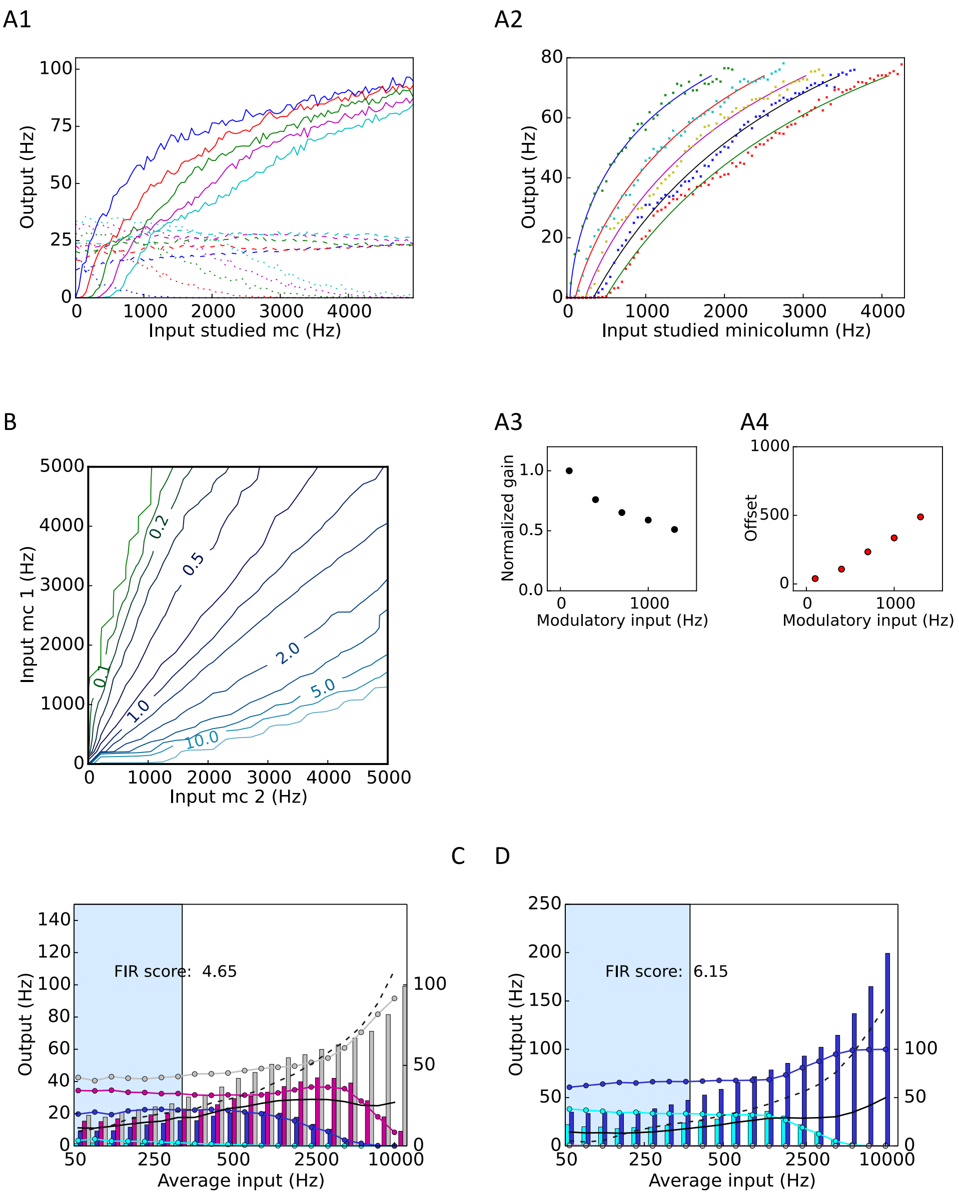

Figure 5.23: Test results Model A tuned. Parameters changed from those seen in Table 4.2: IPSC$_{bas\_pyr}$ = 12.0 pS, EPSC$_{pyr\_bas}$ = 0.005 pS, EPSC$_{ext\_bas}$ = 0.007 pS, p_{in_bas} = 0.05. A1) IO test. Input to other minicolumns 100, 400, 700, 1000 and 1300 Hz. A2) Hill function fit (Equation 4.4) to data points in A below 75 Hz. A3) Normalized gain for curves in B1, between 5 Hz and 75 Hz. A4) Offset for curves in A2. B) Relationships between outputs for two active minicolumns as a function of the input to those minicolumns. C) FIR1234 test. D) FIR1200 test.

5.4.1 Model A large and medium IPSC$_{bas\_pyr}$

First the approach of using a very large value of IPSC$_{bas\_pyr}$ was tested for Model A. This was done to see if the result would be similar to those from Model A1 and A2. An IPSC$_{bas\_pyr}$ of 40.0 pS was set and the network was then tuned manually. This IPSC corresponds to an IPSP size of 7.5 mV. It should however be remembered that the inhibitory reversal potential is – 10 mV, which means the increase in synaptic strength is not completely reflected by the IPSP size.

The results are seen in Figure 5.22 and this approach indeed creates a network that show clear divisive gain change for the IO curve of a studied minicolumn. In the IO test it should especially be noted that gain change is present also for higher input magnitudes. There is also a subtractive shift present, but it is considerably smaller compared to model versions using a smaller IPSP. This model version also gives the so far longest range over which the network pass the FIR1234 test. There is also stable output relations over a larger range of input magnitudes, also for more dissimilar inputs (Figure 5.22 C). It is also clear that average is kept constant over a very large range of input magnitudes for both the IO and FIR tests. The FIR score results Table 5.4 can be compared to other versions of Model A and further confirms that this model version is a good approximation to the normalization model.

However since the biological realism of this mechanism should rightly be questioned, further investigations focus on other features that might contribute to a more normalizing network. For that purpose Model A was instead tuned within a narrower parameter range, to get a perhaps more realistic performance measure. The results from using $\text{IPSC}_{\text{bas\_pyr}}$ = 12.0 pS (corresponding to an IPSP size of 3.9 mV) and tuning other synaptic weights manually are seen in Figure 5.23. This network will be referred to as “Model A tuned”. It shows better performance compared to Model A with standard parameters, but worse than when a very large IPSP is used. The gain change present in the IO test is weaker and accompanied by a clear subtractive shift. The FIR graphs show the same features seen in the previous model versions using smaller IPSPs but also a slightly prolonged region of constant output relations. Model A tuned is in the remainder of this section used for evaluation different changes to Model A that might give a more normalizing network.

Table 5.4: FIR scores for different versions of Model A. All scores are averaged over 5 different random seeds. Green: best scores.

	Model A standard	**Model A, tuned $\text{IPSP}_{\text{bas\_pyr}}$ = 12.0 pS**	**Model A, large $\text{IPSP}_{\text{bas\_pyr}}$ = 40.0 pS**
FIR1234	2.59 ± 0.14	4.17 ± 0.24	**10.16 ± 2.36**
FIR1234 average	3.24 ± 0.22	17.84 ±- 1.2	**21.16 ± 2.25**
FIR1200	3.73 ± 0.26	5.83 ± 0.39	**10.20 ± 0.69**
FIR1200 average	3.73 ± 0.26	7.29 ± 0.42	**26.38 ± 1.83**

5.4.2 Variability in the network

One difference between using a single cell and a group of cells as the computational unit is that for a group of cells is it possible to have variability in parameters such as neuron size/excitability (here cell membrane capacitance C_m) and input to neurons. It was therefore investigated if changing the variability in the network might affect performance. First using an increased and decreased standard deviation for pyramidal and basket neuron capacitance and input variance was tested. The random distribution from which these values were drawn during setup was as described in Section 4.2 a truncated normal distribution. The result seen in Table 5.5 show that there seems to be a weak trend towards better FIR scores when an increased variance is introduced. It is not a very large difference and could be an effect of slightly worse average control, but when considering output relations for the FIR test graph in Figure 5.24, this indicate that the network show a slightly stronger normalizing effect.

Table 5.5: Effect of increased standard deviation in the distributions for cell membrane capacitance C_{m_rsd} and driving input, $Input_{rsd}$. The same values were used for pyramidal and basket cells. All FIR scores are averaged over 5 different random seeds. Standard deviation is given as relative standard deviation (RSD). Green: best values.

Membrane capacitance RSD	Input RSD	FIR1234 score	FIR1234 average score
0.01	0.01	4.35	**18.82**
0.10	0.10	4.05	17.00
0.20	0.20	4.45	16.44
0.25	0.50	**5.176**	14.99

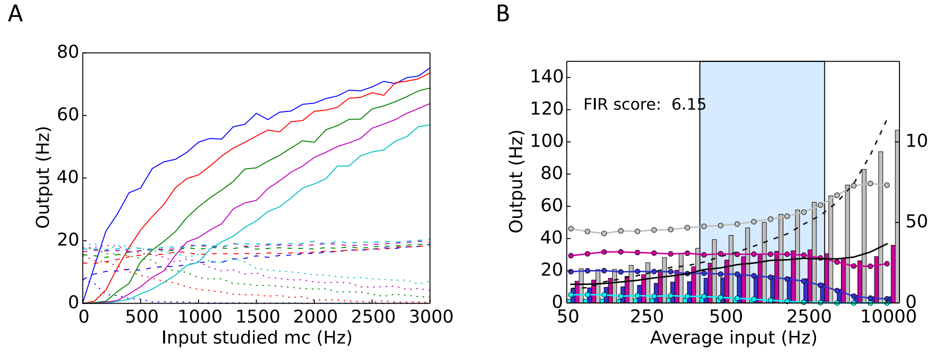

Figure 5.24: Results from IO and FIR1234 tests when increasing variability in neuron C_m and input. For producing these graphs C_{m_rsd} was increased from 0.1 to 0.25, and $Input_{rsd}$ from 0.1 to 0.5 for both pyramidal and basket cells. A) IO test. B) FIR1234 test.

Variability in connections

Since there seemed to be small positive effect from increased variability in the network, a second way of doing this was also investigated: Instead of using a fixed number of incoming connections for each neuron in the postsynaptic population, the number of incoming connections was allowed to vary as described in Section 4.2.1 The results of this approach are seen in Figure 5.25 A and compared to changing the variance of neuronal membrane capacitance and input this does affect network processing in a similar but more pronounced way. The difference is even more pronounced for larger input magnitudes and modulatory inputs as seen in Figure 5.25 B. It is also seen in the FIR graphs that including variability in the number of incoming connections for Model A creates a network better at keeping stable output relations. This can also be seen in Figure 5.25 E as the straighter contour lines in the plot of output relations. It can also be noted that this network have more pronounced WTA behavior, as seen by the closer spacing between contours in Figure 5.25 D. However as seen on the IO graph this does not create a perfectly normalizing network but a subtractive shift is still evident. Figure 5.26 illustrates one reason this mechanism results in improved normalization.

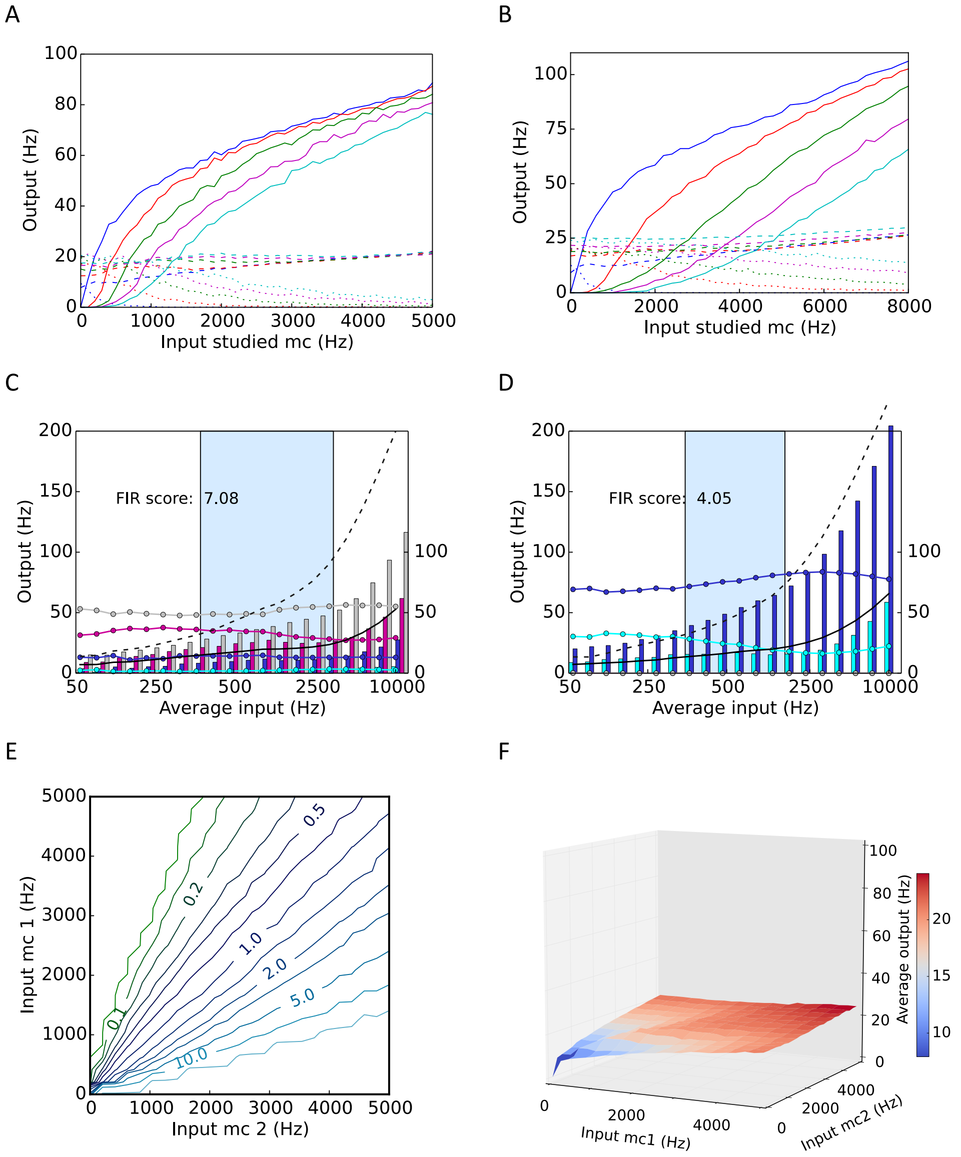

Figure 5.25. Test results for Model A tuned, using a variable number of incoming connections for postsynaptic neurons. A) IO test. Modulatory input: 100, 400, 700, 1000 and 1300 Hz. B) IO test. Modulatory input: 200, 1200, 2200, 3200 and 4200 Hz. Note the gain change present also for higher input magnitudes and larger modulatory inputs. C) FIR1234 test. D) FIR1200 test. E) Relationship between outputs for a network with two active minicolumns, as a function of input to the respective minicolumns. F) As E but average activity in the network as a function of input to the two minicolumns. Note that when variability in incoming connections is introduced, the network is considerably better at keeping output relations stable.

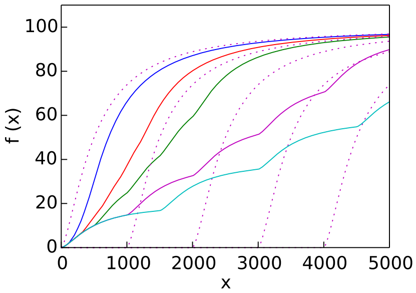

Figure 26: Illustration of one mechanism by which increased variability in the network creates a more normalizing network. Here f(x) (continuous lines) is defined as the average of five Hill functions (Equation 4.4), f_0, f, f_2, f_3 and f_4. For all f_i Rmax = 100, n = 1.5 and sigma = 500. However how they are affected by inhibition is varied, by multiplying a subtractive modulatory input with coefficients 0, 1, 2, 3 and 4 before each function is evaluated. The combined function f(x) is plotted for different amounts of subtractive inhibition, I = 100, 300, 500, 1000 and 1500. Note how this creates a modulatory effect very similar to normalization, even though each f_i is affected purely subtractively by inhibition. The purple dotted curves are the curves that are averaged to compute f(x) for I = 1000.

5.4.3 Recurrent connectivity in minicolumns and basket cell population

The effect of changing the strength of the recurrent connectivity in pyramidal and basket cell population was also investigated. This was done by computing an average FIR1234, as well as FIR1234 average score, for a range of connection probabilities between 0.0 and 0.5. The result is seen in Figure 5.27 E and F. IO and FIR graphs were also plotted for the largest probability of a recurrent connection (0.5) to look for qualitative changes. Those graphs are also seen in Figure 5.27.

It can be seen that increased strength in the recurrent connectivity within minicolumns is coupled to an increased in range of approximately stable average. There is also a slightly better FIR1234 score for recurrent connectivity lower and higher than the one used during this thesis ($p_{pyr\_pyr} = 0.2$). However this is not enough to create a normalizing network. Changing the strength of the recurrent connectivity in the basket cell population does not seem to affect the FIR1234 score much; but there is a marked decrease in the range of approximately stable average, which is perhaps not very surprising.

5.4.4 Adaptation and more realistic firing patterns

Since a more complex neuron model with adaptation is used in the original attractor memory model, it was also investigated whether using model neurons with more realistic firing characteristics would affect the networks normalizing ability. For this purpose the original model neurons in Model A were replaced with adaptive IAF neurons, and the neuronal parameters changed to values shown to implement firing characteristics closely resembling those of the original Hodgkin-Huxley model neurons (see Table 4.2). Synaptic weights was tuned to give similar PSP sizes as Model A tuned or Model A standard.

The results are seen in Figure 5.28 and the network characteristics are very similar to Model A tuned. There are some differences in the dynamics if we study a single input vector - such as a tendency for population bursts in the minicolumns for low magnitude inputs (not shown) - but the results from the FIR tests as well as the IO test are qualitatively very similar to a network with regular IAF neurons.

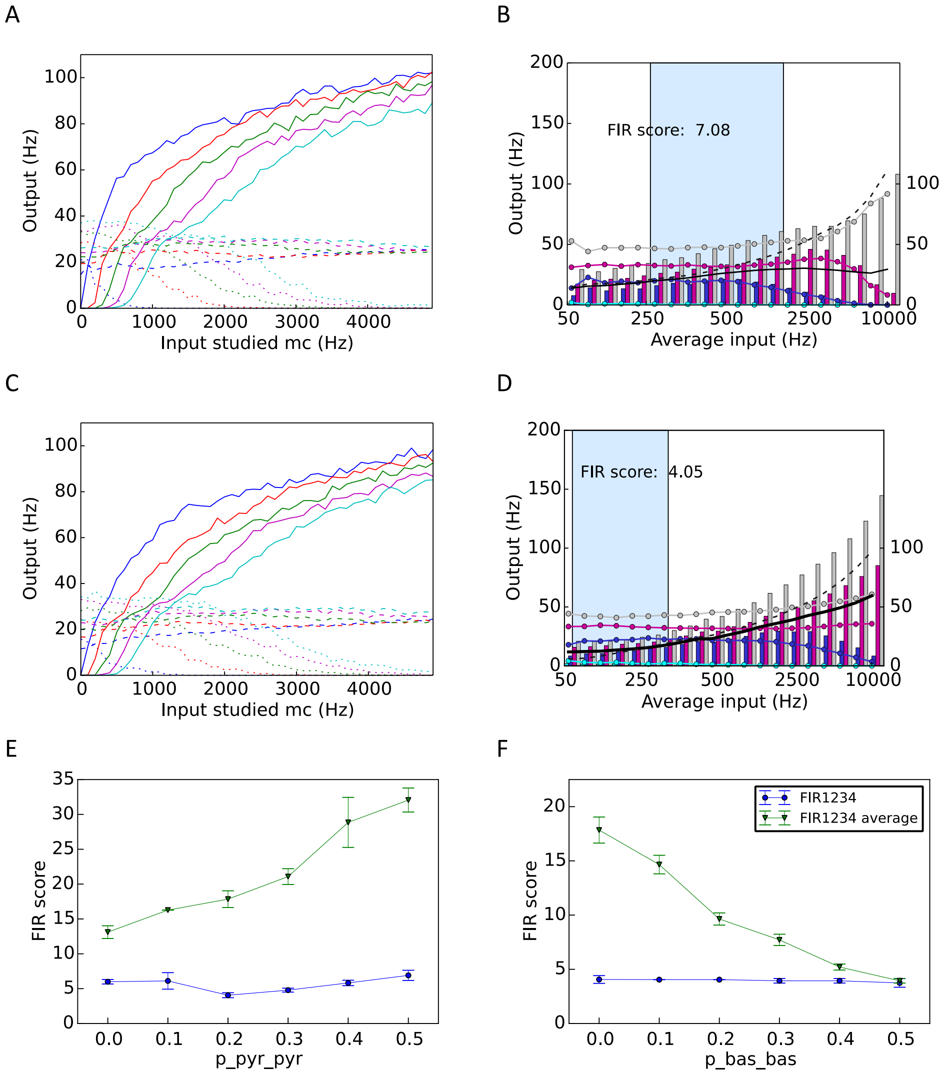

Figure 5.27: IO and FIR test for Model A tuned, with increased recurrent connectivity in pyramidal and basket cell populations. No other parameters are changed. A, B) IO test and FIR1234 test with recurrent connectivity in pyramidal cell populations increased from 0.2 to 0.5. C, D) IO test and FIR1234 test with recurrent connectivity in basket cell population increased from 0.0 to 0.5. Note the qualitative similarity to Model A tuned. E) Result of FIR1234 and FIR1234 average test as a function of p_{pyr_pyr}. F) As E but for p_{bas_bas}.

The FIR ratings for the network using adaptive IAF neurons are shown in Table 5.5, and those are also similar to the results for Model A tuned. In addition it is seen that also for this network the performance on the FIR tests gets worse if a parameter setting similar to that for Model A standard was used. Using a smaller IPSP removed most of the gain change and increased the offset (data not shown). This indicate that the larger increase in membrane variance for increased modulatory input created by a larger IPSP is a very important mechanism also in this model version. The effect of inhibition on the IO curve of a single model neuron was also investigated and the results were similar to those for a regular

IAF neuron (not shown). The conclusion is that the more realistic firing patterns does not seem to introduce any new normalizing mechanisms in the hypercolumn module.

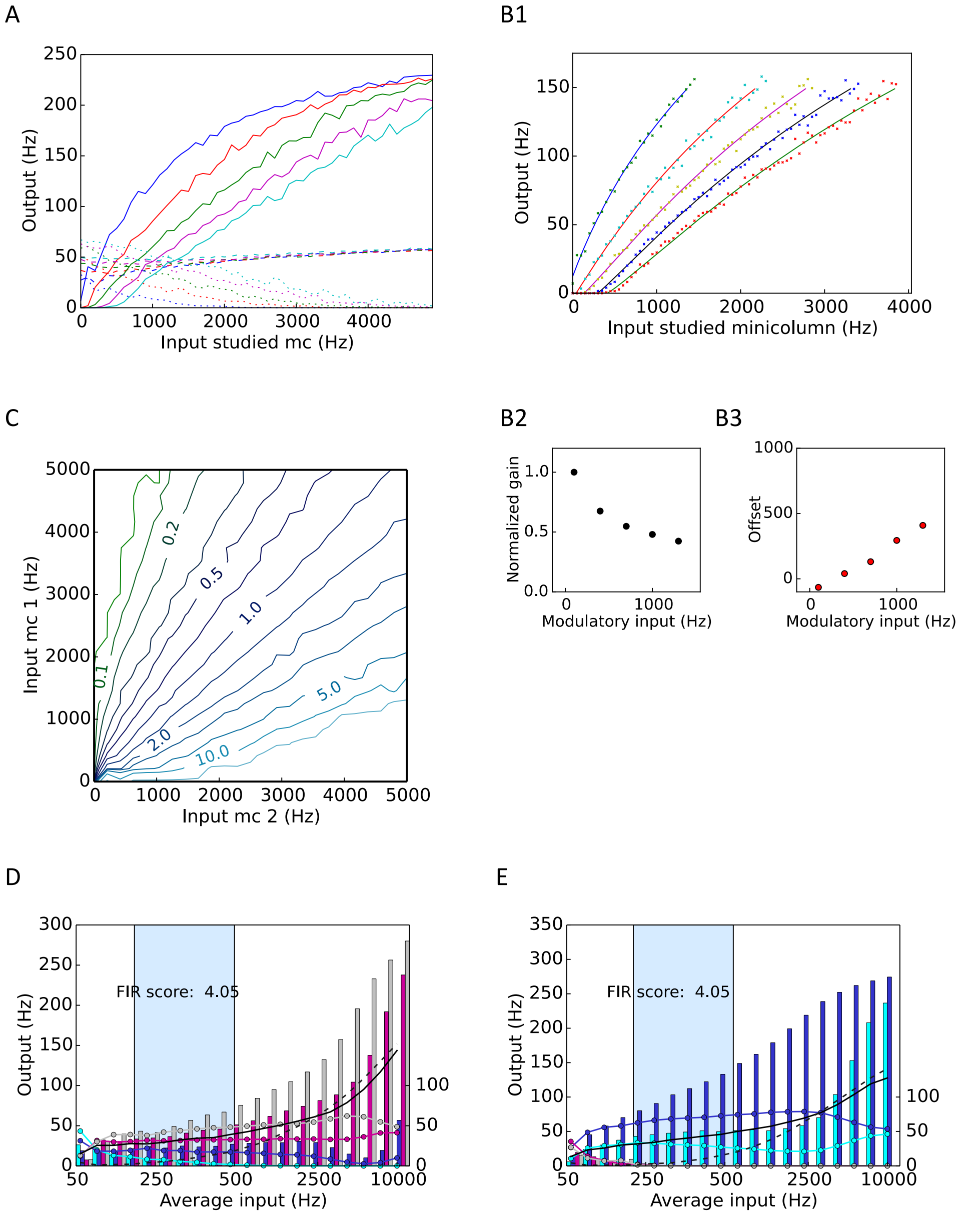

Figure 5.28: Test results when using adaptive IAF-neurons with more realistic firing characteristics. Model neuron parameters are seen in Table 4.2. Other parameter changes relative to Model A standard: EPSC$_{pyr\_bas}$ = 0.009 pS, EPSC$_{ext\_bas}$ = 0.005 pS, IPSC$_{bas\_pyr}$ = 15 pS, p$_{in\_bas}$ = 0.05, noise$_{pyr}$ = 2000 Hz, noise$_{bas}$ = 3500 Hz. *A) IO test. B1) Hill function fit to data points in A below 150 Hz. B2) Gain for curves in B1. B2) Offset for curves in B1. C) Relationship between outputs for a network with two active minicolumns as a function of the input to those minicolumns. D) FIR1234 test. E) FIR1200 test. Note the similarity Model A tuned.*

Table 5.5: FIR ratings for Model A using adaptive IAF neurons and Model A tuned with variability in the number of incoming synaptic connections (here denoted VAR). All FIR scores are averaged over 5 different random seeds. Green: best scores. "+ VAR": Variability in incoming connections added.

	Model A tuned Adaptive neurons IPSC_{bas_pyr} = 15.0 pS	**Model A Adaptive IPSC_{bas_pyr} = 15.0 pS**	**Model A tuned + VAR**
FIR1234	4.28 ± 0.30	3.06 ± 0.0	**7.11 ± 1.58**
FIR1234 average	5.67 ± 0.39	4.28 ± 0.30	**7.92 ± 0.42**
FIR1200	4.28 ± 0.30	2.66 ± 0.0	**5.83 ± 0.39**
FIR1200 average	5.67 ± 0.39	5.35 ± 0.0	**7.28 ± 0.42**

5.5 Model B - Short-term depression

Short-term depression have been found to have a large impact on the response curve of a single neuron (Rothman et al 2009). Because of this it was investigated if adding short-term depression to the driving synapses would have a similar effect. First the same parameters was used as in the tuned version of Model A (results not shown), but some changes proved necessary to optimize performance. Since the effect of the inhibition was enhanced by the short-term depression, the feed-forward input to the basket cells was reduced to 40 % of its previous strength and IPSC_{bas_pyr} and EPSC_{ext_bas} reduced to 80 % and 70 % of their previous values.

The test results for Model B tuned are seen in Figure 5.29. There is a marked qualitative difference from all previous network versions: A clear decrease in R_{max} for increased modulatory inputs. There is also a marked gain change for low input magnitudes and no clear subtractive shift. All those are features typical for the theoretical output gain model. However the point on the horizontal axis where R_{max} is reached, is also shifted towards higher output magnitudes. This indicate that the network also implement normalizing features.

The results from the FIR tests and from studying a network with only two minicolumns, seen in Figure 5.29 B, confirm that the network is quite good at keeping output relations constant for input rates under 2500 Hz. However also for this network, the range of approximate normalization is smaller for more dissimilar inputs. When the input magnitude increase further the distinction between outputs is lost in the same way seen for the theoretical output gain model.

The FIR test also show a clear resemblance to the theoretical output gain model: The average first increase but is then slowly decreasing and clearly bounded. The relationships between the outputs also follow the typical pattern of first diverging and then converging, until all distinction is lost between outputs. But it should be noted that this happens slower than for the theoretical output gain model. As seen in Table 5.6 the results of the FIR tests are very good. Something that should be noted compared to the previous model versions is that the STD network have a more pronounced problem with keeping distinction between similar outputs. This can also be seen in Figure 5.30 B, which show a pattern qualitatively different to previous model versions.

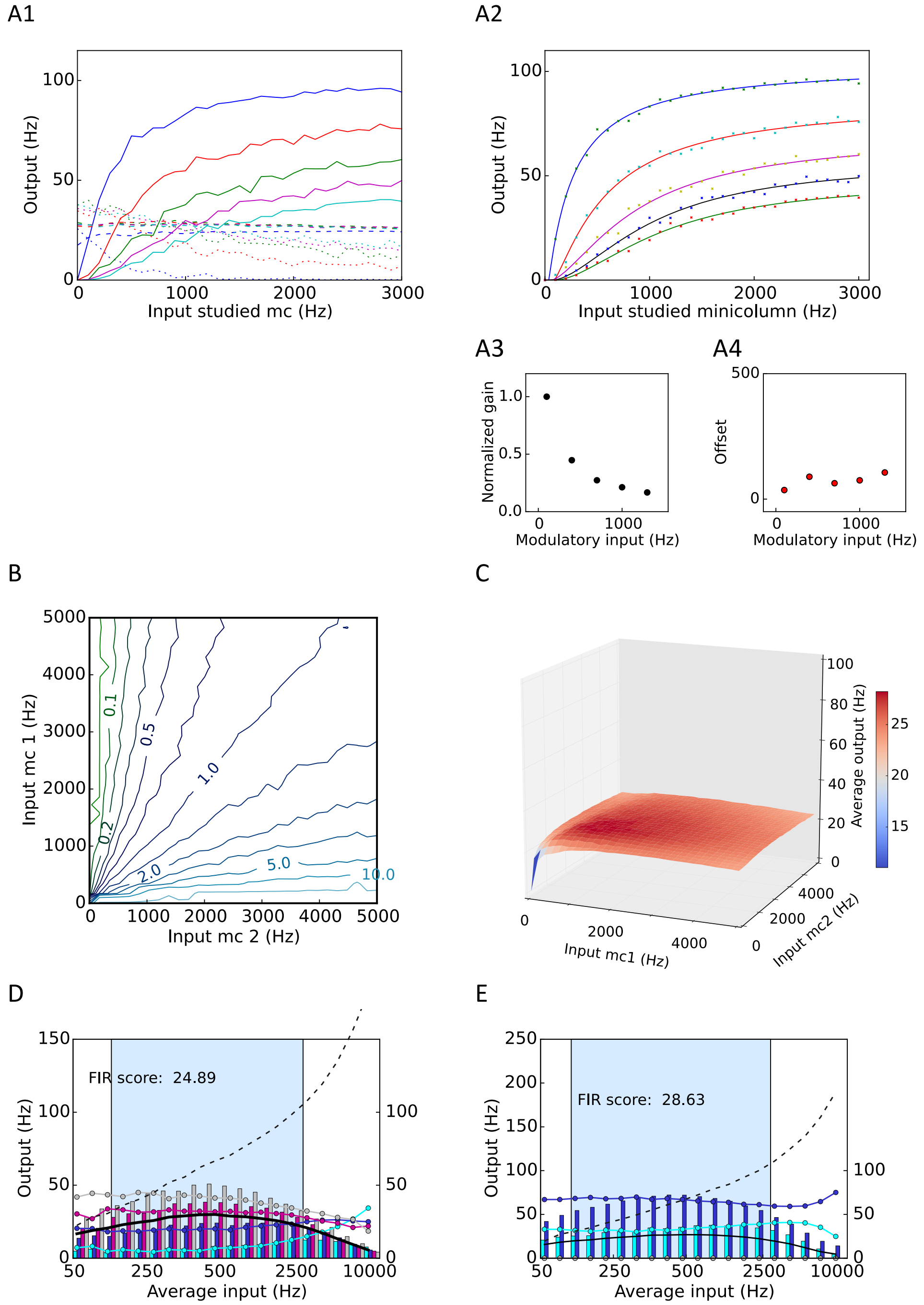

Figure 5.29. Test results Model B tuned. Differences in parameters from standard parameters in Table 4.2: $IPSC_{bas\_pyr}$ = 12.0 pS, $EPSC_{pyr\_bas}$ = 0.005 pS, $EPSC_{ext\_bas}$ = 0.004 pS, p_{in_bas} = 0.02. A1) IO test. Input other minicolumns 100, 300, 700, 1000 and 1300 Hz. A2) Sigmoid fit to data points in A. A3) Normalized gain between 5 and 75 % of R_{max} for each curve in A2. A4) Offset for curves in A2. B) Relations between outputs for a network with two active minicolumns as a function of input to those minicolumns. D) As B but average output. D) FIR1234 test. E) FIR1200 test.

Synaptic depression on pyramidal-pyramidal synapses

In the original attractor memory model synaptic depression is included on all pyramidal-pyramidal synapses; that is also for the recurrent connections within the minicolumns. This was also tested and gives results very similar to those in the previous section (not shown) but the FIR scores were slightly lower. The reason for this is probably that, as shown in Section 5.4.3, stronger recurrent connectivity within minicolumns tend to increase the range of stable average.

Smaller IPSP and short-term depression

To see if the effect from short-term depression is similar when using a smaller IPSP, the FIR1234 and IO tests were repeated using IPSC$_{bas\_pyr}$ = 2.6 pS (IPSP 1.1 mV). The results are seen in Figure 5.30. The similarity to the output gain model is still evident, but the network now has more trouble keeping the average and output relations stable. Increasing inhibition strength did not improve performance (data not shown).

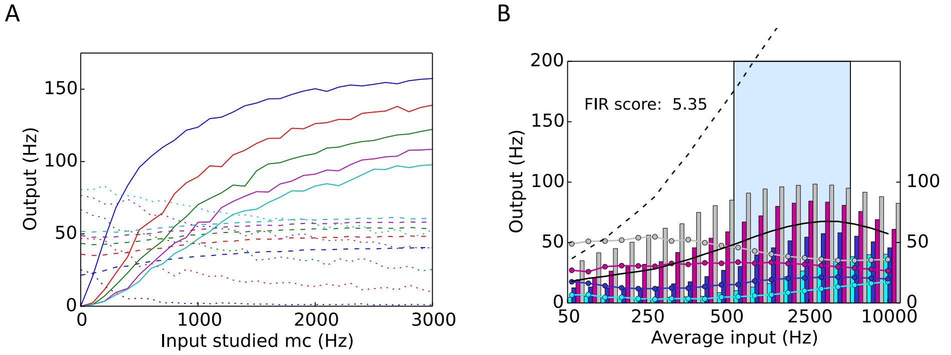

Figure 5.30: IO test and FIR test for Model B using a smaller IPSC$_{bas\_pyr}$ of 2.6 pS. Parameter changes compared to the network in the previous section: EPSC$_{ext\_bas}$ = 0.006 pS, EPSC$_{pyr\_bas}$ = 0.005 pS, and p$_{in\_bas}$ = 0.05. Note that average control is considerably worse and that output relations vary with input magnitude in B.

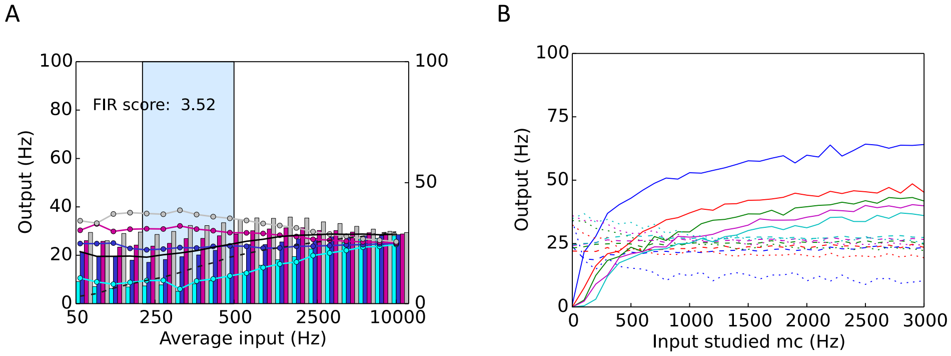

Figure 5.31: Short-term depression does not produce as good results if the variability in incoming connections is removed from the network. FIR1234 and IO test for Model B with the same parameters as in Figure 5.29, but using a fixed number of incoming connections to postsynaptic neurons. A) FIR1234 test. B) IO test. Modulatory inputs: 100, 400, 700 and 1300 Hz.

Variability in the network and short-term depression.

It is interesting to see how important the variability in in incoming connections to postsynaptic neurons is for a network implementing gain change through synaptic depression. Therefore Model B was also tested using a fixed number of incoming connections to postsynaptic neurons. As seen in Figure 5.31 the performance is not nearly as good, which is further supported by the FIR scores in Table 5.6, where scores for all versions of Model B are presented. Thus using a variable number of incoming connections seem to be crucial for good performance also when gain change is implemented through short-term depression.

Table 5.6: FIR test results for different versions of Model B. All FIR scores are averaged over 5 different random seeds. "+ VAR": variability in incoming connections added, "- VAR": no variability in incoming connections. Green: best scores.

	STD **IPSP$_{bas\_pyr}$ = 12.0 pS** **+ VAR**	**STD** **IPSP$_{bas\_pyr}$ = 12.0 pS** **- VAR**	**STD** **IPSP$_{bas\_pyr}$ = 2.6 pS** **+ VAR**
FIR1234	**17.41 ± 9.5**	3.18 ± 0.86	3.15 ± 0.18
FIR1234average	**36.03 ± 4.21**	7.96 ± 0.85	10.76 ± 0.0
FIR1200	**26.02 ± 8.20**	4.06 ± 0.36	3.24 ± 0.22
FIR1200average	**34.90 ± 2.42**	10.80 ± 0.96	11.40 ± 0.49

5.6 Additional tests

Since this project was of an exploratory character, several other experiments were conducted, that were judged to be less enlightening to report, or very similar to results already described. None of them gave any marked improvement of normalization. Those worth mentioning are:

- Including recurrent connections between all minicolumns in the hypercolumn. This tended to, not surprisingly, impair distinction between outputs.
- Including recurrent connections between minicolumns receiving similar input. This was less detrimental but tended to impair distinction between the two largest outputs.
- Changing the connection probabilities between the minicolumns and the basket cell populations, as well as between the basket cell population and minicolumns. A connection probability close to 100 % impaired normalization, otherwise no marked effect.
- Using current based instead of conductance based synapses. No marked effect on normalization.
- Using alpha shaped, current or conductance based synapses. No marked effect on normalization.
- Using different neuronal parameters for the IAF or adaptive IAF neuron model but tuning PSPs to the same values. No marked effect on normalization.
- Using a population of pyramidal cells instead of parrot neurons for input to Model B. No marked effect on normalization, but a slightly noisier output.

6 Discussion and conclusions

The discussion chapter is divided into two parts: A discussion part where the results are summarized and possible conclusions; causes for observed effects; and the possible use of this type of system are discussed more extensively, and a shorter conclusion part that summarizes the most important conclusions to be drawn from the study.

6.1 Discussion

This section begin by discussing the results for Model A with standard parameters. The differences between the feed-back and the feed-forward circuits as well as what limits these model versions normalizing ability are summarized. Thereafter the three mechanism found to give a more normalizing network are discussed: noise, variability in neuron excitability and connections and short-term depression. Last the methodology is evaluated and alternative approaches are considered.

6.1.1 Model A standard parameters

If the standard values for synaptic weights, or values close to those, are used and each neuron receives the same number of incoming connections, there were only slight differences between different implementations of Model A. The modulatory effects implemented by the feed-forward inhibitory circuit in Model A1, the feed-back circuit in Model A2 or by the combined circuits in Model A are all best described as qualitatively most similar to the subtractive inhibition model. All versions show the characteristic subtractive shift of the IO curve for increased modulatory input. Qualitative features seen in the FIR graphs further supports that the modulatory effect is primarily subtractive. Those features are:

- Average output not stable for increased input magnitudes.
- For fixed input relations the relationships between the outputs show a strong dependence on input magnitude.
- Also for constant input magnitude, average output might vary depending on the relationships between the inputs.

The network models tendencies for WTA behavior only for medium input magnitudes can as well be explained by the subtractive inhibition model if also the sigmoidal shape of the basket cell population's response curve is taken into account. The constructed mathematical model of the system gave primarily one important insight considering why it is hard to keep relationships between outputs for increased input magnitude: Since all minicolumns receive the same amount of inhibition (from the common inhibitory population), but their driving input will grow with different factors, it is not possible to tune inhibition to keep all outputs constant. That the basic assumption this model was based upon, a subtractive shift of the IO curve for a minicolumn for increased inhibition, is a good description of network functionality was also shown by plotting the output from a single minicolumn receiving different amounts of excitatory drive to the pyramidal and basket cell populations (Figure 5.3, Figure 5.17).

Normalizing features

That the response curve of a minicolumn is shown to be primarily subtractively shifted for increased modulatory inputs explains why these model versions are not very good at normalization. But there were in fact some weak normalizing effects observed. First for low input magnitudes and small modulatory inputs, some gain change was present in addition to the subtractive shift. This was more pronounced the smaller output magnitudes that were studied. Second in the FIR test graphs, the model versions were showing a bit slower divergence of output relations and thus an approximate normalization over a slightly larger range than for the theoretical subtractive inhibition model.

This gain change was also seen for a single neuron and minicolumn detached from the network which shows it is not a network effect. It can instead be explained by the increased variance in the input created by both increased input rate and the added inhibition. However this effect is not large enough to give a pure divisive rescaling of the IO curve for standard parameters.

It was also noted that the network was better at keeping stable relationships between the outputs if a small increase in average was allowed, which is the reason the best results on the FIR test is not seen for the same inhibition strength as the best average control. This depends on that an increasing average counteract diverging output relations, since the gain of a sigmoidal response curve is strictly decreasing except for very low input magnitudes; the minicolumns receiving a smaller increase in excitatory drive during the FIR tests, will thus have a larger gain than the minicolumns receiving a larger increase in excitatory drive. This in fact highlights another important insight from studying these model versions: Keeping average constant does not necessarily create a normalizing network.

Feed-forward versus feed-back inhibition

One question we set out to answer was if there were differences in the normalizing ability of the feed-forward and feed-back inhibitive circuits. The results do as mentioned show that the difference in normalizing ability is very small. However some qualitative differences were found that could be important to consider if the aim is to create an approximately normalizing hypercolumn. The two most pronounced differences are:

- The feed-forward circuit is better at keeping a stable average, while average is strictly increasing for the feed-back circuit.
- The feed-back network is less sensitive to tuning because of the feed-back loop creating a dynamic balance in the firing rates of the excitatory and inhibitory populations.

To use both types of inhibitory circuits in concert can thus be motivated, and by choosing the relative strength of feed-forward versus feed-back inhibition one of those features can be emphasized. The differences found in this work that a feed-forward circuit is more powerful for controlling average activity, while a feed-back circuit is more dynamically adapting could be a possible explanation for why both these circuits are present in cortex.

In conclusion these network model versions can even though qualitatively closer to the subtractive inhibition model, be used to approximate normalization in a spiking neural network for a limited range of input magnitudes and input relations. However that normalization works considerably worse for more disparate input relations should be taken into account. That is the reason we are not stating that these network versions “are normalizing” for the low input region, since that is only true for a subset of input relations. If a hypercolumn module with these parameters should be used to normalize output in a spiking neural network, is thus dependent on that requirements for accuracy is not too high. Using a hypercolumn with these parameters could e. g. be good enough if the important point is to have an approximately constant average but the quotient between the first and second largest input is not crucial.

6.1.2 The role of noise and IPSP size

For Model A, A1 and A2 the effect of increasing IPSP size was to shift the network from the subtractive inhibition towards the normalization model. This effect was found to be more pronounced the larger IPSPs that were used. The reason for this was touched upon in the previous section: A computational unit receiving inhibition will need larger excitatory drive to reach a certain output rate. This will create a larger variability in incoming spikes at rheobase (at least if as in this study the input has Poisson statistics) and variability will also be further increased by the variability added by the inhibitory drive. The effect of an increased modulatory input in the network can thus be separated into two parts:

- A decrease in average membrane potential.
- An increase in input, and thus membrane potential variability.

As seen in previous studies, see e. g. Chance et al (2002), the combination of a subtractive shift, and increased variance can if properly balanced create a pure divisive shift in a neurons IO curve. It is for all networks versions in this thesis as well as for a single neuron or minicolumn seen that an increased modulatory input is tied to increased neuron membrane variability. When smaller IPSPs are used this effect is weak. However when a larger IPSP is used, it is magnified and for the “large IPSP” networks studied here, large enough to create a network that is better described the normalization model than any of the other theoretical models.

There are several studies showing that variable amounts of balanced noise has the potential of being an important neuronal mechanism for gain change (Ayaz & Chance 2009, Ly & Doiron 2009, Prescott & De Koninck 2003). It has also been suggested that this is the biological mechanism that gives the “divisive” part in the normalization equation (Carandini & Heeger 2012). However there have to our knowledge not been any computational studies where this noise creating gain change is shown to be generated *intrinsically* in the network. This should thus be considered one of the more interesting findings of this study.

This is also to our knowledge also the first complete “normalizing circuit” built from the “gain change via noise” principle. To get normalization gain change is needed, but it also has to be an appropriate amount at the right time. There are other studies that model normalizing circuits such as Ayaz & Chane (2009) but then the “normalizing signal” (which is for that study a separate noise signal) is generated external to the model.

The main argument against that the “large IPSP versions“ of the hypercolumn model is that the IPSP and thus IPSC needed to create a sufficiently large effect are very large. The IPSP size for the network model versions that perform really well is about 10 times the average found by Thomson et al. (2002) and could thus not be considered biologically realistic. This is of course a reason to doubt the validity of the mechanism; and that a large enough increase of noise can be created in a single hypercolumn to give a normalizing circuit. Another important parameter is that the driving input is modelled as Poisson processes which gives this increase in variance with input magnitude. Since all input is modelled naturally the statistics of this input will be very important.

It should however be remembered that the mechanisms works also for smaller IPSPs (just not as good) and could thus maybe work in concert with other mechanisms – such as synaptic depression - to give normalization in a cortical hypercolumn. And the approximate normalization performed by this network found for more realistic parameter values might be at play for tasks when the range over which the input magnitude is varied is quite small.

It should also be remembered that we here study a single hypercolumn module with very few minicolumns. Using a very small network can sometimes result in a need to compensate by using unrealistic parameter values. In conclusion further studies are needed to assess if this is an artificial effect only possible to create in this specific network model or a viable mechanism for normalization in a biological hypercolumn. One interesting line of future studies would be to investigate this mechanism in a larger network, with connectivity similar or different to the original attractor memory network and which reproduce spike statistics typically seen in cortex.

6.1.3 Variability in the network

It was seen in the results section that the model versions performed better the larger amount of variability that was present in model neuron capacitance, input to neurons in a minicolumn, and number of incoming connections. For neuron capacitance and input the effect was quite small. This is most likely partly due to that a truncated normal distribution was used, so there was a limit to the amount of variability that was introduced. It is also the case that even if there e. g. is a variability in the feed-forward input to basket cells, since each pyramidal cell gets input from several basket cells, the variability in the amount of inhibition for a pyramidal cell will be smaller than the input variability for the basket cells.

By allowing the number of incoming connections to each neuron to vary it is possible to create a larger variability in excitatory and inhibitory input to pyramidal cells. The result was a binomial distribution of inhibitory as well as excitatory inputs. Some neurons could thus e. g. receive triple the average excitatory input and a third of the average inhibitory input or no inhibitory input at all. Using a variable number of incoming connections accordingly have a larger impact on network functionality. The network model versions including this variability was considerably better at keeping output relations stable, even though this did not translate to a perfectly normalizing IO curve. It was also shown by a simple computational example that variability in inhibition can give rise to a marked normalizing effect also if the response curve of a single computational unit is affected purely subtractively by a modulatory input.

That this type of variability in the network affect can have so large impact on the network functionality should be considered the second important finding in this study. This shows that maybe not all modulatory effects need to come from a single neuron implementing this effect intrinsically, which previous normalization and gain control studies have been focused on. Maybe the answer to the elusive normalization phenomena in cortex cannot be found only in a single neuron but we need to lift the gaze? The effect on network processing of this type of variability is even more pronounced when short-term depression is included in the network. An interesting direction of future studies could be thus to study the effects and biological realism of a more structured (not as here random) variability in the network. This could both create an inhibitory population able to provide linear (or in fact arbitrary shape of the IO curve) inhibition over a larger range of input magnitudes and create normalizing effects in the network by the principle illustrated in Figure 26. This could possibly be done modeled on how different neuron sizes affect recruitment and inhibitability of motor neurons (Henneman et al. 1965).

These results also highlights that the typical choice of using neurons of similar size, may result in inability to capture some important phenomena. An important consideration is however of course, that there are several studies showing that the output from a single neuron is normalized; and that a normalizing output on population level can thus not fully explain normalization in cortex. However this is in relation to an external stimuli and not afferent inputs. An interesting possibility to consider could thus be if the normalized response could in fact be the effect of input from an "upstream" normalizing group of neurons.

6.1.4 Short-term synaptic depression

Adding short-term synaptic depression on driving synapses was what had the largest effect on the network processing, shifting its modulatory effect from subtractive inhibition to output gain. This was seen for all versions of Model B, but for the version using a variable number of incoming connections as well as a quite large IPSP the most interesting results were seen. In addition to the reduced maximum output and gain change for low inputs typical for the output gain model, there was also seen a shift in the half-max of the response curve. This is a feature typical for the normalization model, and this network can thus be said to implement a mix of normalization and output gain. Although qualitatively most similar to the output gain model, especially for higher input magnitudes, this network was able to keep average output and output relations over the longest range of all model versions, as measured by the FIR test. Also here it should however be noted that the model is less normalizing for certain input relations; for this model version this is in contrast to previous model versions more *similar* inputs.

This is the third significant result of this study: Variability in the network tend to shift the effect from synaptic depression previously shown to be present for single neurons (Rothman et al. 2009) towards the normalization model. This is thus another example of how using a group of neurons results in different processing compared to when a single neuron is studied. However also this model version is most accurately described as "output gain approximating normalization" over a certain range of input magnitudes and input relations. This because of the clear qualitative similarities to output gain seen for the IO and FIR tests.

But synaptic depression could indeed be used to create an approximately normalizing circuit for low and medium input magnitudes. This is an important result since it could enable a biologically realistic implementation of such a circuit for future spiking network models. Could it also be a good enough approximation to explain the normalizing effects seen in biological cortical circuits? Between 50 and 1500 Hz input magnitude, which in an increase in input magnitude with a factor of 30, this network quite closely approximate the normalization model. For an increase in input magnitude with a factor of 10 it is a very good approximation. Although normalization for some neural circuits such as e. g. on the retina can handle a much larger change in input magnitude than that, it is a considerable range. There was not time within this study to study more extensively whether this range might be enough to explain some of the normalizing effects observed in primate cortex. This is thus believed to be an important future line of study. It would also be interesting to investigate how blocking short-term depression might affect the normalizing ability of a biological cortical circuit.

A computational study by Ayaz and Chance (2009) also showed that the modulatory effect was shifted from output gain to normalization when mutual inhibition was included between a normalization pool and a second pool of cortical neurons. So an interesting line of study could also be to investigate if something similar might be possible in this type of network. The large difference in the modulatory effect seen when synaptic depression was added - which is indeed present at biological pyramidal to pyramidal synapses - indicate that this feature might have large implications on information processing in cortex and could be a plausible explanation for normalizing effects.

6.1.5 Alternative approaches

A different approach to this problem could have been using a more detailed neuron model, such as the Hodgkin Huxley multicompartmental models used in the original attractor memory network (Lundqvist et al. 2006). The results of this thesis indicate that this would probably not have made any difference since the main limitation to the networks normalizing ability comes from that the model neuron IO curve is a) Strictly increasing and b) Affected in a subtractive way by moderate amounts of inhibition. Both these conditions are also true for a Hogdkin-Huxley neuron model of a pyramidal cell as shown by e. g. Rothman et al. (2009) and Chance et al. (2002). Changing the neuron type is thus not likely to affect the reason why the structure of the hypercolumn module does not per se create a normalizing network. However there are slight differences in how the response curve of an Hodgkin-Huxley neuron and an IAF neuron are affected by noise (Chance et al. 2002), so it could be interesting to see if the "noise mechanism" would work better or worse for a different neuron model.

Another choice made was to study a standalone hypercolumn module, which of course cannot represent the dynamics of a larger network. The question we should ask is if a hypercolumn embedded in the original attractor memory network might be more normalizing. An indication this might not be the case is that also in the attractor memory model a large increase in the activity of each "winning" minicolumn is seen if the amount of excitation to those are increased (Lundqvist et al. 2006). This is similar to what was found for several network versions in this study and indicate the "constant" activity in each hypercolumn in the original attractor network is in fact quite dependent on that each minicolumn in an active pattern receive an approximately constant amount of excitation. This indicate that also a model comprising a large network of hypercolumns do have similar dynamics concerning keeping average constant. That the WTA effect of the hypercolumn in the attractor memory network is reproduced here also indicates that the dynamics is similar. The most important limitation of using a single hypercolumn module is probably that it does not produce the same type of realistic spike statistics and since spike statistics are important for the "noise mechanism" that could be worth investigating further.

It would also have been possible to use an even simpler rate based model, but this was not deemed accurate enough since it can e. g. not capture the gain change of a neuronal response curve for increased input variability. This do in hindsight seem to be a good choice since this specific mechanism was in fact shown to have a large impact on network processing.

The FIR test have shown to be very useful for studying the network model versions modulatory effects more in detail, and get a deeper understanding of different reasons a network might be more or less normalizing. However it have because of the network models dependence on input relations shown to not be a perfect way to compare different model versions; this since it only studies a single input vector. The results are still considered useful, but it would be helpful to develop a more holistic measurement of the range over which a network can be considered normalizing.

6.2 Conclusions

This thesis has investigated which modulatory effects that can be created in a highly structured recurrent spiking neural network, modelling layer 2/3 of a cortical hypercolumn. This means a network composed of several intrinsically connected groups of cells (minicolumns) with a common inhibitory pool creating feed-back/feed-forward inhibition or a combination of those. The results show that the structure of the hypercolumn does not per se create a normalizing network. Since the response curve of a single minicolumn in the network when using standard parameters is affected primarily subtractively by inhibition, it is not possible to fulfill normalization criteria of keeping both output relations and average stable for any larger range of input relations. The modulatory effect is then instead more accurately described by the subtractive inhibition model.

However there are mechanisms in this type of circuit that can shift the modulatory effects towards the normalization model. In the hypercolumn module following normalizing mechanisms were found:

- An increase in input variance coupled to larger modulatory inputs.
- Variability in the network, either in neuronal properties and input, or in the number of incoming connections from presynaptic neurons.
- Short-term depression on the driving synapses.

The first mechanism is widely proposed as one of the mechanism giving rise to the divisive part of a normalizing circuit, but this is (to our knowledge) the first time this noise is shown to arise intrinsically in a spiking neural network model. That it here require moving outside the boundaries of IPSP sizes observed in cortex does not necessarily mean the mechanism itself is not valid. The second mechanism show that using a group of connected neurons as the computational unit can indeed create dynamics not present for a single neuron. This indicates that to improve our understanding of the mechanisms for normalization in biological neural circuits we may need to lift the gaze from the modulatory effects that can be implemented by a single neuron and also study larger computational units. Adding short-term depression resulted in a qualitatively different modulatory effect, best described as output gain. However when combined with network variability this gives a network which can approximate normalization over a large range of input magnitudes and input relations. Short-term depression combined with the other two mechanisms could thus be a candidate for how normalization is implemented in cortex.

However to know if these mechanisms can constitute a biologically realistic explanation for normalization in biological nervous systems, further studies are needed. An interesting direction of those studies would be to further investigate if and how short-term depression might be combined with other mechanisms to create a more fully normalizing network; or if the effect seen is in fact large enough to explain some of the normalizing effects seen in primate cortex. Further investigations should also be directed towards studying if there are other ways the increased noise can be created in the hypercolumn module apart from using very large IPSPs and towards structured variability in networks. It would also be useful to study if and how the functionality of this type of hypercolumn module differ when it is a part of a larger network.

References

Abbott LF, Chance FS. 2005. Drivers and modulators from push-pull and balanced synaptic input In *Cortical Function: A View from the Thalamus*, ed. VA Casagrande, RW Guillery, SM Sherman, pp. 147-55. Amsterdam: Elsevier Science Bv

Adrian ED, Zotterman Y. 1926. The impulses produced by sensory nerve endings: Part 3. Impulses set up by Touch and Pressure. *The Journal of physiology* 61: 465-83

Ayaz A, Chance FS. 2009. Gain Modulation of Neuronal Responses by Subtractive and Divisive Mechanisms of Inhibition. *Journal of Neurophysiology* 101: 958-68

Beck JM, Latham PE, Pouget A. 2011. Marginalization in Neural Circuits with Divisive Normalization. *J. Neurosci.* 31: 15310-19

Brette R, Gerstner W. 2005. Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of Neurophysiology* 94: 3637-42

Brette R, Rudolph M, Carnevale T, Hines M, Beeman D, et al. 2007. Simulation of networks of spiking neurons: A review of tools and strategies. *J. Comput. Neurosci.* 23: 349-98

Bruederle D, Petrovici MA, Vogginger B, Ehrlich M, Pfeil T, et al. 2011. A comprehensive workflow for general-purpose neural modeling with highly configurable neuromorphic hardware systems. *Biological Cybernetics* 104: 263-96

Brunel N. 2000. Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J. Comput. Neurosci.* 8: 183-208

Cannon B. 2014. KVA Molecular Frontiers Symposium and Youth Forum, 20-21 May 2014.

Carandini M, Heeger DJ. 1994. Summation and division by neurons i primate visual cortex. *Science* 264: 1333-36

Carandini M, Heeger DJ. 2012. Normalization as a canonical neural computation. *Nat. Rev. Neurosci.* 13: 51-62

Carandini M, Heeger DJ, Movshon JA. 1997. Linearity and normalization in simple cells of the macaque primary visual cortex. *J. Neurosci.* 17: 8621-44

Carandini M, Heeger DJ, Senn W. 2002. A synaptic explanation of suppression in visual cortex. *J. Neurosci.* 22: 10053-65

Chance FS, Abbott LF, Reyes AD. 2002. Gain modulation from background synaptic input. *Neuron* 35: 773-82

Cortical structure. 2014. Cortical organization - columnar structure. Kaohsiung Medical University Wiki, http://wiki.kmu.edu.tw/index.php/. Accessed 2014-11-22

Cybenko G. 1989. Approximation by superpositions of a sigmoidal functions. *Mathematics of Control Signals and Systems* 2: 303-14

DeFelipe J, Markram H, Rockland KS. 2012. The neocortical column. *Front. Neuroanat.* 6: 2

Douglas RJ, Martin KAC. 2004. Neuronal circuits of the neocortex. *Annu. Rev. Neurosci.* 27: 419-51

Eppler JM, Helias M, Muller E, Diesmann M, Gewaltig M-O. 2008. PyNEST: A Convenient Interface to the NEST Simulator. *Frontiers in neuroinformatics* 2: 12-12

Gabbiani F, Krapp HG, Hatsopoulos N, Mo CH, Koch C, Laurent G. 2004. Multiplication and stimulus invariance in a looming-sensitive neuron. *Journal of Physiology-Paris* 98: 19-34

Gewaltig M-O, Morrison A, Plesser H. 2012. NEST by Example: An Introduction to the Neural Simulation Tool NEST In *Computational Systems Neurobiology*, ed. N Le Novère, pp. 533-58: Springer Netherlands

Ghosh-Dastidar S, Adeli H. 2009. Spiking Neural Networks. *Int. J. Neural Syst.* 19: 295-308

Henneman E, Somjen G, Carpenter DO. 1965. Excitability and inhibitability of motoneurons of different sizes. *Journal of neurophysiology* 28: 599-620

Herculano-Houzel S. 2009. The human brain in numbers: a linearly scaled-up primate brain. *Front. Hum. Neurosci.* 3: 11

Hodgkin AL, Huxley AF. 1952. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology* 117: 500-44

Holt GR, Koch C. 1997. Shunting inhibition does not have a divisive effect on firing rates. *Neural Comput.* 9: 1001-13

Hübel DH, Wiesel TN. 1959. Receptive fields of single neurons in the cat's striate cortex. *Journal of Physiology*: 574-91

Isaacson JS, Scanziani M. 2011. How Inhibition Shapes Cortical Activity. *Neuron* 72: 231-43

Izhikevich EM. 2003. Simple model of spiking neurons. *IEEE Trans. Neural Netw.* 14: 1569-72

Kostal L, Lansky P, Rospars JP. 2007. Neuronal coding and spiking randomness. *Eur. J. Neurosci.* 26: 2693-701

Kouh M, Poggio T. 2008. A canonical neural circuit for cortical nonlinear operations. *Neural Comput.* 20: 1427-51

Lansner A. 2009. Associative memory models: from the cell-assembly theory to biophysically detailed cortex simulations. *Trends Neurosci.* 32: 178-86

Lansner A. 2014. Personal communication.

Litwin-Kumar A, Doiron B. 2012. Slow dynamics and high variability in balanced cortical networks with clustered connections. *Nat. Neurosci.* 15: 1498-505

Louie K, Khaw MW, Glimcher PW. 2013. Normalization is a general neural mechanism for context-dependent decision making. *Proc. Natl. Acad. Sci. U. S. A.* 110: 6139-44

Lundqvist M, Herman P, Lansner A. 2011. Theta and Gamma Power Increases and Alpha/Beta Power Decreases with Memory Load in an Attractor Network Model. *Journal of Cognitive Neuroscience* 23: 3008-20

Lundqvist M, Herman P, Lansner A. 2013. Effect of Prestimulus Alpha Power, Phase, and Synchronization on Stimulus Detection Rates in a Biophysical Attractor Network Model. *J. Neurosci.* 33: 11817–24

Lundqvist M, Rehn M, Djurfeldt M, Lansner A. 2006. Attractor dynamics in a modular network model of neocortex. *Netw.-Comput. Neural Syst.* 17: 253-76

Luo SX, Axel R, Abbott LF. 2010. Generating sparse and selective third-order responses in the olfactory system of the fly. *Proc. Natl. Acad. Sci. U. S. A.* 107: 10713-18

Ly C, Doiron B. 2009. Divisive Gain Modulation with Dynamic Stimuli in Integrate-and-Fire Neurons. *PLoS Comput. Biol.* 5: 12

Ma WJ, Beck JM, Latham PE, Pouget A. 2006. Bayesian inference with probabilistic population codes. *Nat. Neurosci.* 9: 1432-38

Maass W. 1997. Networks of spiking neurons: The third generation of neural network models. *Neural Netw.* 10: 1659-71

Marsland. 2009. *Machine learning - An algorithmic perspective*.

McCulloch WS, Pitts W. 1943. A logical calculus of the ideas immanent in nervous activity *Bulletin of Mathematical Biophysics* 5?: 115-33

Mountcastle VB. 1997. The columnar organization of the neocortex. *Brain* 120: 701-22

Murphy BK, Miller KD. 2003. Multiplicative gain changes are induced by excitation or inhibition alone. *J. Neurosci.* 23: 10040-51

Ohshiro T, Angelaki DE, DeAngelis GC. 2011. A normalization model of multisensory integration. *Nat. Neurosci.* 14: 775-U281

Olsen SR, Bhandawat V, Wilson RI. 2010. Divisive normalization in olfactory population codes. *Neuron* 66: 287-99

Olshausen BA, Field DJ. 2004. Sparse coding of sensory inputs. *Curr. Opin. Neurobiol.* 14: 481-87

Papadopoulou M, Cassenaer S, Nowotny T, Laurent G. 2011. Normalization for Sparse Encoding of Odors by a Wide-Field Interneuron. *Science* 332: 721-25

Ponulak F, Kasinski A. 2011. Introduction to spiking neural networks: Information processing, learning and applications. *Acta Neurobiol. Exp.* 71: 409-33

Prescott SA, De Koninck Y. 2003. Gain control of firing rate by shunting inhibition: Roles of synaptic noise and dendritic saturation. *Proc. Natl. Acad. Sci. U. S. A.* 100: 2076-81

Purves D, Augustine GJ, Fitzpatrick D, Hall WC, LaMantia A-S, et al. 2008. *Neuroscience fourth edition*. Sunderland, M. A., USA: Sinauer Associates, Inc.

Rabinowitz NC, Willmore BDB, Schnupp JWH, King AJ. 2011. Contrast Gain Control in Auditory Cortex. *Neuron* 70: 1178-91

Reynolds JH, Heeger DJ. 2009. The Normalization Model of Attention. *Neuron* 61: 168-85

Ringach DL. 2009. Spontaneous and driven cortical activity: implications for computation. *Curr. Opin. Neurobiol.* 19: 439-44

Rothman JS, Cathala L, Steuber V, Silver RA. 2009. Synaptic depression enables neuronal gain control. *Nature* 457: 1015-18

Schwartz O, Simoncelli EP. 2001. Natural signal statistics and sensory gain control. *Nat. Neurosci.* 4: 819-25

Shadlen MN, Newsome WT. 1998. The variable discharge of cortical neurons: Implications for connectivity, computation, and information coding. *J. Neurosci.* 18: 3870-96

Sherman SM, Guillery RW. 1998. On the actions that one nerve cell can have on another: Distinguishing "drivers" from "modulators". *Proc. Natl. Acad. Sci. U. S. A.* 95: 7121-26

Silver RA. 2010. Neuronal arithmetic. *Nature reviews. Neuroscience* 11: 474-89

Steratt D, Graham B, Gillies A, Willshaw D. 2011. *Principles of computational neuroscience*. New York: Cambridge University Press.

Stufflebeam R. 2014. Introduction to Neurons, Synapses, Action Potentials, and Neurotransmission. In *The Mind Project* *http://www.mind.ilstu.edu/curriculum/, accessed 2014-10-27*

Thomson AM, West DC, Wang Y, Bannister AP. 2002. Synaptic connections and small circuits involving excitatory and inhibitory neurons in layers 2-5 of adult rat and cat neocortex: Triple intracellular recordings and biocytin labelling in vitro. *Cereb. Cortex* 12: 936-53

Tsodyks M, Pawelzik K, Markram H. 1998. Neural networks with dynamic synapses. *Neural Comput.* 10: 821-35

van Atteveldt N, Murray MM, Thut G, Schroeder CE. 2014. Multisensory Integration: Flexible Use of General Operations. *Neuron* 81: 1240-53

Waydo S, Kraskov A, Quiroga RQ, Fried I, Koch C. 2006. Sparse representation in the human medial temporal lobe. *J. Neurosci.* 26: 10232-34

7 Appendix

7.1 Mathematical analysis of the theoretical models

In this sections a mathematical analysis which show that the features discussed for the normalization, output gain and subtractive inhibition model in Section 4.3.3, hold for all values of, n, σ and R_{max}. It also gives a more detailed analysis of why they occur.

Normalization

Consider the output $\overline{y} = [y_1, y_2, .. y_n]$ from a normalizing network with inputs $\overline{x} = [x_1, x_2, .. x_n]$:

$$\bar{y} = (y_1, y_2, .. \ y_m) = R_{max} \frac{R_{max}}{\sigma^n + k\sum_{j\neq i} x_j^n + x_i^n} \cdot (x_1^n, x_2^n, .. \ x_m^n) \tag{7.1}$$

The average output for a specific input vector is:

$$\langle y \rangle \ = \frac{1}{m} R_{max} \frac{\sum_j x_j^n}{\sigma^n + k\sum_{j\neq i} x_j^n + x_i^n} \tag{7.2}$$

For $\sum_j x_j^n > \sigma^n$ the sigmoidal term will approach one and we get

$$\langle y \rangle \ \approx \frac{R_{max}}{m} \tag{7.3}$$

that is the average activity approach a constant independent of n, input magnitude and the relationships between the inputs. We now consider the relationships between the outputs when we keep the input relations constant that is $\overline{x} = [c_1a, c_2a, .. c_ma]$. Since all inputs are divided by the same denominator we can write the output in Equation 7.1 as

$$\bar{y} = \frac{1}{C(a)} (x_1^n, x_2^n, .. \ x_m^n) = \frac{1}{C(a)} ((c_1 a)^n, (c_2 a)^n, .. \ (c_m a)^n) \tag{7.4}$$

The relationship between two outputs y_i and y_j will thus be:

$$\frac{y_i}{y_j} = \frac{a^n {c_i}^n}{a^n {c_j}^n} = \frac{{c_i}^n}{{c_j}^n} \tag{7.5}$$

That is a normalizing network will keep the relationships between the outputs constant independent of R_{max}, σ and the magnitude of the input as was illustrated in Figure 4.4. The relationships between the outputs do however depend on n - a larger n makes the network go towards winner take all dynamics, while a smaller n tend to keep the relationship between the outputs the same as that between the inputs (this is identically true for n = 1). It can also be noted that in contrast to subtractive inhibition one computational unit will never be completely silenced, even if its activity can be very low.

Output gain

Considering the output of the network for the output gain model that will be:

$$\bar{y} = R_{max}\frac{1}{(1+k\sum_j c_j a)}\left(\frac{(c_1a)^n}{\sigma^n + c_1a^n}, \frac{(c_2a)^n}{\sigma^n + c_2a^n}, \dots \frac{(c_ma)^n}{\sigma^n + c_ma^n}\right) \tag{7.6}$$

When $(c_i a)^n \ll \sigma^n$ for all c$_i$ the numerators in the second term will grow faster than the denominator in the first term, and we will thus have an increasing average. On the other hand when $(c_i a)^n \gg \sigma^n$ for all c$_i$, the second term will for all outputs saturate, while the denominator of the first term continue growing (albeit slowly for a small k) and thus the average will now decrease with increasing input magnitude. There is thus not a stable average as for normalization, but it will be bounded. The same is true for the output from individual units which will not reach their max firing rate. If we consider the relationship between the inputs we have:

$$\begin{aligned}\frac{y_i}{y_j} &= R_{max}\left(\frac{1}{(1+k\sum_k c_k a)}\frac{(c_ia)^n}{\sigma^n + c_ia^n}\right)\Big/ R_{max}\left(\frac{1}{(1+k\sum_j c_j a)}\frac{\left(c_ja\right)^n}{(\sigma^n + c_ja^n)}\right)\\ &= \left(\frac{(c_ia)^n}{\sigma^n + c_ia^n}\right)\Big/\left(\frac{\left(c_ja\right)^n}{(\sigma^n + c_ja^n)}\right)\end{aligned} \tag{7.7}$$

When $(c_i a)^n \ll \sigma^n$ for all c$_i$, the small inputs will first grow slower than the larger ones, partly since $ac_i < ac_j$ and partly since the derivative of the sigmoidal term (for n>1) is increasing for low input magnitudes, and a larger input thus have a larger gain. However when the derivative of the sigmoidal term start to decrease the small inputs will grow faster than the large ones, creating a gradually smaller difference between them. When $c_i a^n \gg \sigma^n$ for all c$_i$ all the outputs will be saturated and we get:

$$\frac{y_i}{y_j} \approx 1 \tag{7.8}$$

In conclusion the relationships between the outputs will for the output gain model in the beginning depend on the relationships between the inputs but approach one if the input magnitude is increased enough.

Subtractive inhibition

Repeating the analysis for subtractive inhibition we start by considering the average output:

$$\bar{y} = R_{max}\left(\begin{gathered}\frac{\left(c_1a - k\sum_j c_j a\right)^n}{\sigma^n + \left(c_1 a - k\sum_j c_j\, a\right)^n}, \frac{\left(c_2a - k\sum_j c_j a\right)^n}{\sigma^n + \left(c_2 a - k\sum_j c_j\, a\right)^n},\\ \dots \frac{\left(c_ma - k\sum_j c_j\, a\right)^n}{\sigma^n + \left(c_m a - k\sum_j c_j\, a\right)^n}\end{gathered}\right) \tag{7.9}$$

Since

$$(c_i a - k\sum_j c_j a) = a(c_i - k\sum_j c_j) \tag{7.10}$$

Where the second term is independent of a, this will give two types of outputs. If $(c_i - k\sum_j c_j) > 0$ the output from that unit will grow as a sigmoid however shifted to the right on the horizontal axis. If on the other hand $(c_i - k\sum_j c_j) < 0$ that output unit will be silenced no matter the input. Since all outputs are either growing or zero this will give a monotonically increasing average output that saturates when $a(c_i - k\sum_j c_j) \gg \sigma^n$ for all i. The average activity will then be:

$$\langle y \rangle \approx \frac{R_{max}\, m_{active}}{m} \tag{7.11}$$

Where m_{active} is the number of non-silenced units. It should however be noted that this stable average output is only reached when all non-silenced units have reached their max firing rate, that is when any differences between the inputs are no longer perpetuated to the outputs. This means similar to output gain this type of modulation work best over a small range of input magnitudes. For subtractive inhibition it is also interesting to consider the case where we have a “starting input” before inhibition gets started (as this will later be the case for the network models). If all inputs are the same it is trivial to tune the inhibition to keep a constant average as seen in Figure 4.5 A. However when they are not it is more complicated as seen can be seen in Figure 4.5 B.

Considering the relations between the outputs we have:

$$\frac{y_i}{y_j} = R_{max}\frac{\left(c_i a - k\sum_j c_j a\right)^n}{\sigma^n + \left(c_i a - k\sum_j c_j\, a\right)^n} \Bigg/ R_{max}\frac{\left(c_j a - k\sum_j c_j a\right)^n}{\sigma^n + \left(c_j a - k\sum_j c_j\, a\right)^n}, \tag{7.12}$$

If $\left(c_i a - k\sum_j c_j a\right)^n \ll \sigma^n$ for all i, this simplifies to:

$$\begin{aligned} \frac{y_i}{y_j} &\approx \frac{\left(c_i a - k\sum_j c_j a\right)^n}{\sigma^n} \Bigg/ \frac{\left(c_j a - k\sum_j c_j a\right)^n}{\sigma^n}, \\ &= \frac{a^n(c_i - k\sum_j c_j)^n}{a^n(c_j - k\sum_j c_j)^n} \\ &= \frac{(c_i - k\sum_j c_j)^n}{(c_j - k\sum_j c_j)^n} \end{aligned} \tag{7.13}$$

Which means for small inputs the relationship between the outputs will be decided by the relationship between the inputs, k and n, and stay approximately stable with increasing input magnitude. When $a^n(c_i - k\sum_j c_j)$ grows however, this approximation is no longer valid and since the derivative of the sigmoids in Equation 4.4 for n > 1, is strictly decreasing for larger input magnitudes, the smaller outputs will eventually grow faster than the larger ones diminishing the difference between them. In the limit where all non-silenced units reach max firing rate there will no longer be a discernible difference between the outputs.

7.2 Python code

The network model is structured in the form of a hypercolumn class that keep track of all nodes such as neurons, Poisson generators, and devices to record network activity. There are no separate classes for minicolumn and basket cell populations, but those are represented as lists of nodes. The build and connect methods of the hypercolumn class define the network structure and are seen in Listing 7.1. The complete hypercolumn class as well as parameters and functions for performing some of the simulations can be downloaded from GitHub https://github.com/spacemir/normalization. To run the scripts NEST, as well as the python packages NumPy, SciPy, and Matplotlib must be available. Versions used are Python 2.7.5, NEST 2.2.2, NumPy 1.8.2, SciPy 0.13.3 and Matplotlib 1.2.1.

Listing 7.1 Python code showing the build and connect methods of the hypercolumn class.

```python
class HC:
    """Hypercolumn class with variable number of minicolumns"""

    def build(self):
        # Each neuron will have its own input rate,
        # and therefore an individual Poisson generator
        self.inputs = [None]*(self.params["n_minicolumns"]+1)
        for i in range(self.params["n_minicolumns"]):
            poisson_generators = Create("poisson_generator",self.params["N_E"])
            self.inputs[i] = poisson_generators
        self.inputs[-1] = Create("poisson_generator",self.params["N_I"])
        self.pyr_noise_input = Create("poisson_generator")
        self.bas_noise_input = Create("poisson_generator")
        self.spikedetectors = Create("spike_detector",\
                      self.params["n_minicolumns"]+1)
        self.raster_spikedetector = Create("spike_detector")
        self.voltmeters = Create("voltmeter",self.params["n_minicolumns"]+1)

        #create populations of excitatory cells
        for n in range(self.params["n_minicolumns"]):
            mc = Create(self.params["neuron_model"], self.params["N_E"],\
                        params=self.params["pyramidal_params"])
            self.all_pyr = self.all_pyr + mc
            self.minicolumns.append(mc)

        #create populations of inhibitory cells
        self.basket_cells = Create(self.params["neuron_model"],\
        self.params["N_I"], params=self.params["basket_params"])
        self.randomize_network_and_inputs()
        self.built = True

    def connect(self):
        if not self.built:
             self.build()

        #create all synapse models
        CopyModel("static_synapse", "pyr_external",
             "weight":self.params["epsp_ext_pyr"],"delay":self.params["d"]})
        CopyModel("static_synnapse", "pyr_recurrent",
             {"weight":self.params["epsp_pyr_pyr"], "delay":self.params["d"]})
        CopyModel("static_synapse", "bas_external",
             {"weight":self.params["epsp_ext_bas"],"delay":self.params["d"]})
        CopyModel("static_synapse", "bas_recurrent",
             {"weight":self.params["ipsp_bas_bas"],"delay":self.params["d"]})
        CopyModel("static_synapse", "pyr_inhibitory",
             {"weight":self.params["ipsp_bas_pyr"],"delay":self.params["d"]})
        CopyModel("static_synapse", "bas_excitatory",
             {"weight":self.params["epsp_pyr_bas"],"delay":self.params["d"]})

        # choice between variable number of outgoing or incoming connections
        if self.params.["mult"]:
             connect = RandomDivergentConnect
        else:
             connect = RandomConvergentConnect

        # Connections..
        self.raster_neurons = []
        for i,mc in enumerate(self.minicolumns):
             Connect(self.inputs[i],mc, model = "pyr_external")
             connect(mc, mc, int(self.params["p_pyr_pyr"]*self.params["N_E"]),\
                      model="pyr_recurrent", options = {'allow_multapses':\
                      self.params["mult"], allow_autapses':self.params["aut"]})
             connect(mc, self.basket_cells,int(self.params["p_pyr_bas"]*
                      self.params["N_E"]),model="bas_excitatory",\
                      options={'allow_multapses':self.params["mult"]})
            connect(self.basket_cells, mc, int(self.params["p_bas_pyr"]*\
                      self.params["N_I"]), model="pyr_inhibitory", \
                      options={'allow_multapses':self.params["mult"]})
            DivergentConnect(self.pyr_noise_input, mc,
                      self.params["epsp_noise_pyr"],  self.params["d"])
            # .. to device
             self.raster_neurons += mc[0:self.params["n_cells_to_raster"]+1]

        if self.params["p_mc_mc"]>0:
             self.mc_mc_connect()

        # .. external noise bc pop, recurrent bc pop
        Connect(self.inputs[-1],self.basket_cells, model="bas_external")
        DivergentConnect(self.bas_noise_input, self.basket_cells,
self.params["epsp_noise_bas"], self.params["d"])
        connect(self.basket_cells, self.basket_cells,int(self.params["p_bas_bas"]*
             self.params["N_I"]),                model="bas_recurrent")
        self.raster_neurons += \
             self.basket_cells[0:self.params["n_cells_to_raster"]+1]

        self.connect_devices()
        self.connected = True

    def __init__(self, params):
        ResetKernel()
        self.params = params    # parameter dictionary
        self.minicolumns = []   # MCs added here during build
        self.inrates = None
        self.setRandomSeeds(params["random_seeds"][3]) #choose a random seed
        self.finished, self.built, self.connected = False, False, False
```

www.kth.se