

**Department of Physics and Astronomy
University of Heidelberg**

Bachelor Thesis in Physics
submitted by

Oliver Julien Breitwieser

born in Essen (Germany)

2011

Investigation of a Cortical Attractor-Memory Network

This Bachelor Thesis has been carried out by Oliver Julien Breitwieser at the

Kirchhoff Institute for Physics in Heidelberg

under the supervision of

Prof. Dr. Karlheinz Meier

Investigation of a Cortical Attractor-Memory Network

When developing neuromorphic hardware, it is of utmost importance to ensure the faithful representation of the network models to be emulated. In this thesis a highly complex cortical attractor-memory network is modified as to comply with the neuron and synapse models available on the FACETS/BrainScaleS Wafer-Scale Hardware. It is shown that all major features in the original model can faithfully be reproduced. These encompass winner-take-all attractor dynamics, pattern completion as well as pattern rivalry, including the attentional blink phenomenon which has been extensively studied by other groups. Furthermore, the effects of hardware-specific distortions on this model are investigated and compensation mechanisms are suggested, where feasible. Additionally, the new model is extended beyond its original structure, with the introduction of non-orthogonal patterns and an application is presented in the form of retinotopic pattern completion.

Untersuchung eines kortikalen Attraktorspeichernetzwerkes

Bei der Entwicklung neuromorpher Hardware ist es außerordentlich wichtig, eine gewissenhafte Repräsentation der zu emulierenden Netzwerkmodelle zu gewährleisten. In dieser Arbeit wird ein hochkomplexes kortikales Attraktorspeichernetzwerk so angepasst, dass es den auf der FACETS/BrainScaleS Wafer-Scale Hardware verfügbaren Neuron- und Synapsenmodellen genügt. Es wird gezeigt, dass alle wesentlichen Charakteristika des Originalmodells getreu reproduziert werden können. Diese umfassen winner-take-all Attraktordynamiken, Mustervervollständigung sowie Musterrivalität, einschließlich des Phänomens des Aufmerksamkeitsblinzeln, welches von anderen Gruppen bereits eingehend erforscht wurde. Des Weiteren werden Einflüsse hardware-spezifischer Verzerrungen auf das hier vorgestellte Modell untersucht und, soweit realisierbar, Möglichkeiten zur Kompensation aufgezeigt. Das neue Modell wird außerdem über seine Originalstruktur hinaus erweitert, zum einen durch die Einführung nicht-orthogonaler Muster, zum anderen durch Vorstellung einer Anwendungsmöglichkeit in Form von retinotopischer Mustervervollständigung.

Contents

1	Introduction	1
1.1	From Existential Considerations to Computational Neuroscience	1
1.2	Beyond the von Neumann Paradigm: FACETS & BrainScaleS	2
1.3	Thesis Outline	2
2	Simulation	3
2.1	Network Model	3
2.1.1	Layout	3
2.1.2	Patterns	5
2.1.3	UP-states	5
2.1.4	Modeling	5
2.1.5	Network Scaling	7
2.1.6	Stimulus	7
2.2	Network Behavior Analysis	9
2.2.1	Data Preprocessing	9
2.2.2	Phase Space Trajectory Projection Plots	11
2.2.3	UP-state Detection	13
2.3	Implementation	17
3	Results	21
3.1	Parameter Fit	21
3.2	Orthogonal Patterns	21
3.2.1	Background Input	23
3.2.2	Pattern Completion	24
3.2.3	Pattern Rivalry	30
3.3	Hardware Imperfections	36
3.3.1	Neuron Loss	36
3.3.2	Synapse Loss	39
3.3.3	Fixed Pattern Noise	43
3.4	Non-Orthogonal Patterns	46
3.5	Retinotopic Pattern Completion	48
4	Discussion and Outlook	53
A	Simulation Parameters	55
A.1	Neuron Model	55
A.2	Synapses	56
A.3	Scaling Formulae	56

B Further Information	57
B.1 Hardware Imperfections in a smaller Network	57
B.2 Diagonal Pattern Distribution Examples	61
C Acronyms and Technical Terms	63
D Bibliography	65

1 Introduction

1.1 From Existential Considerations to Computational Neuroscience

The second law of thermodynamics describes what appears to be a relentlessly destructive principle of nature: as time passes, the universe becomes increasingly disordered. In light of this consideration (while not violating an overall entropy increase), the birth of life, and eventually the emergence of consciousness, seems counterintuitive. The more awe-inspiring it then appears that self-organizing processes inside a single cell, say, a mammalian zygote, give rise to a fully functioning organism gifted with the ability of perceiving and interacting with the surrounding world through a multitude of channels. The self-organization does not stop at birth, though: through complex interactions between billions of nerve cells, throughout the lifespan of their hosts, mammalian brains develop a multitude of astonishing abilities, culminating in the cognitive capabilities of humans, which include feats such as speech, abstract thought and the ever mysterious senses of self and of free will. What we define as “self” and all the things that these “selves” do, such as the writing, reading and understanding of these lines, is the product of – on a cosmic scale – minuscule lump of highly complex self-organized matter. Is it possible for this amazing device of nature to hold a complete understanding of itself? At this point, a definitive answer remains out of sight, although huge steps have already been taken towards achieving this goal.

In comparison to other scientific fields, neuroscience is a relatively young discipline. While rather limited anatomical knowledge can be dated back to ancient Egypt, the first groundbreaking discoveries were made possible by the development of microscopy techniques and the invention of the staining procedure in the late 19th century. Since then, a lot has been discovered about the structure and functionality of individual neurons and their synaptic connections. Several functional areas of the brain, including for example components of the sensorimotor pathway, have since been understood to a large extent. However, despite concentrated efforts, we have yet to find a conclusive microscopic theory of more complex functional areas, such as the ones regulating memory formation and retrieval, inferential reasoning or self-awareness, only to name a few. The reason most certainly lies in the intricate structural complexity of the networks of neurons responsible for these feats, where every single component may play an important functional role and therefore can not be treated with statistical methods alone. It is for this reason that the dawn of the computer age in the mid 20th century held huge promise for the neuroscientific community by offering the possibility of actually simulating detailed models of the scrutinized biological systems. This was the birth hour of computational neuroscience, a discipline which has profited immensely from the vertiginous technological advances in raw computational power that have continuously been taking place for the past decades. Its results are equally impressive, from the design of neural-network controlled robots to the development of complex models of cortical areas.

1.2 Beyond the von Neumann Paradigm: FACETS & BrainScaleS

Even so, the overall computational power of the brain incomparably outmatches even the most powerful of today's machines. Similarly, large-scale neural network simulations quickly run into unsurmountable computational barriers even at sizes several orders of magnitude below a full-scale mammalian brain. The reason is fairly straightforward: While the inherently serial von Neumann architecture of standard computer chips has proven highly efficient from an engineering point of view, it is fundamentally different, if not to say limited, in comparison to the massively parallel architecture of the central nervous system. Even with the simplest neuron and synapse models, the number of computations per unit of time scales quadratically with the size of the network; at the current development rate of processing power, a time where conventional processors will be able to handle significant portions of, say, a human brain, remains out of sight.

The microprocessor revolution has, however, intrinsically gone hand in hand with major technological advances in microelectronics fabrication. Today this allows, in principle, the efficient development of hardware which diverges from standard chip design, possibly veering towards the direction of a more brain-like architecture. This is precisely the idea behind the Fast Analog Computing with Emerging Transient States (FACETS) project, and its successor BrainScaleS (BSS): neuroanatomical and -physiological findings are woven into neural network models which are then emulated on a novel, massively parallel, analog neuromorphic hardware device. This device implements neurons and synapses "in silico" – that is, microcircuits which obey dynamic laws similar to those of their biological archetypes. Together with a great versatility, implemented by design, which allows the emulation of almost arbitrary network structures, the FACETS/BSS hardware offers one more, somewhat implicit advantage: due to the minuscule size of the components involved, time constants are accordingly small, yielding a significant speedup of 10^4 with respect to biological real time. Along with its obvious advantages, this neuromorphic approach also comes with several drawbacks, when compared to conventional simulators. While the latter can be made arbitrarily precise (at the cost of computation time), analog hardware inherently suffers from both design limitations and manufacturing imperfections. Studying the effects of these so-called distortion mechanisms on the dynamics and function of various types of to-be-emulated neural networks is of paramount importance for establishing the adequacy of this landmark approach for neuroscientific research.

This thesis describes ongoing research in understanding the effects of such distortions and, if possible, finding ways of compensating for them. It is concerned with one particular network model and is structured as follows.

1.3 Thesis Outline

The network model itself along with its logical building blocks is described in chapter 2. Furthermore, methods from the analysis module developed to extract and visualize the more abstract features of the network are presented as well. Chapter 3 is concerned with simulation results. It details regular network dynamics, induced behavior such as pattern completion and pattern rivalry as well as investigates how the network reacts to hardware-specific distortion mechanisms. Additionally, it showcases an extension of the network dynamics in the form of non-orthogonal patterns as well as an application in the form of retinotopic pattern completion.

2 Simulation

2.1 Network Model

The network investigated in this thesis is restricted to the layer 2/3 portion of the neocortex and is described in full detail in *Lundqvist et al. [2006]*. It features a twofold columnar structure in which the neurons are organised. The network is comprised of several hypercolumns (HCs), mutually interconnected via excitatory connections. Every HC in itself contains the same number of minicolumns (MCs). The MCs consist of 30 interconnected excitatory pyramidal cells, 2 inhibitory Regular Spiking Non-Pyramidal (RSNP) cells projecting vertically onto the pyramidal cells in the same MC and 1 inhibitory basket cell projecting horizontally onto other MCs within the same HC.

2.1.1 Layout

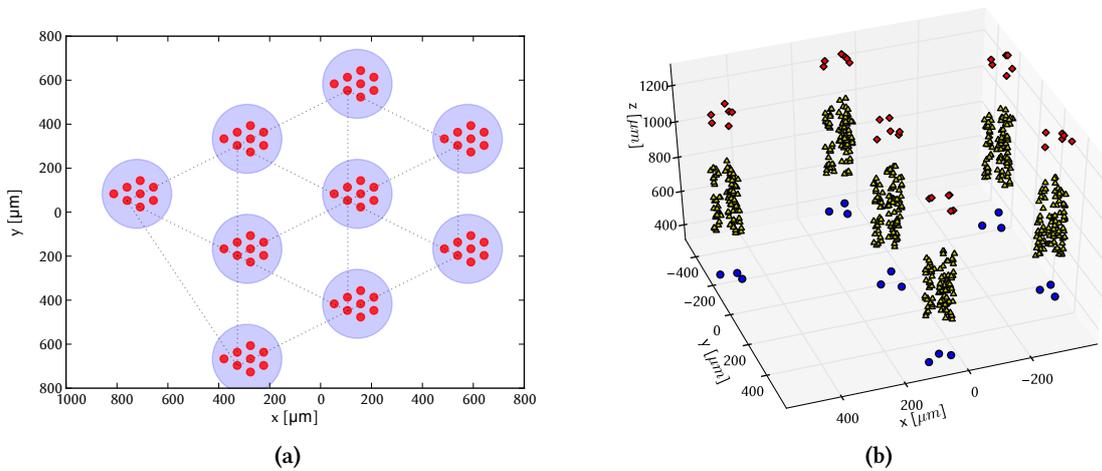


Figure 2.1: (a) Spatial distribution of network components in a setup with 9 HCs (*blue*) on a hexagonal grid with an edge length of $500\ \mu\text{m}$. Within each HC, the 8 MCs (*red*) are distributed in the same manner – the edge length is $60\ \mu\text{m}$. MCs participating in an exemplary pattern (see 2.1.2 *Patterns*) are connected via dotted lines.

(b) Actual 3D distribution of all cells in a smaller network of 6 HCs with 3 MCs each. The cells in an individual MC are distributed within cylindrical shapes with a radius of $15\ \mu\text{m}$ and varying height from bottom to top: basket cells (*blue*, $100\ \mu\text{m}$), pyramidal cells (*yellow*, $400\ \mu\text{m}$) and RSNP (*red*, $100\ \mu\text{m}$). The cylinders of different types of cells are separated by $200\ \mu\text{m}$. Note that in vivo, there is no vertical separation of the three types of neurons. They are all distributed in the complete MC. Hence, the synaptic delays within each MC were taken to be constant (see 2.1.4 *Delays*).

The network represents a subsample from a real cortical network (see *Lundqvist et al. [2006]*), in which each HC is comprised of over 100 MCs. The dimensions of the subsample were left unchanged. Therefore, the HCs are distributed on a hexagonal grid as outlined in figure 2.1a. The distance between the centers of two HCs is $500\mu\text{m}$. Within each HC, the MCs are distributed on a hexagonal grid as well, with a distance of $60\mu\text{m}$ in between. Inside the MCs the cells are uniformly distributed within a vertical cylinder (see figure 2.1b). The network connectivity is outlined

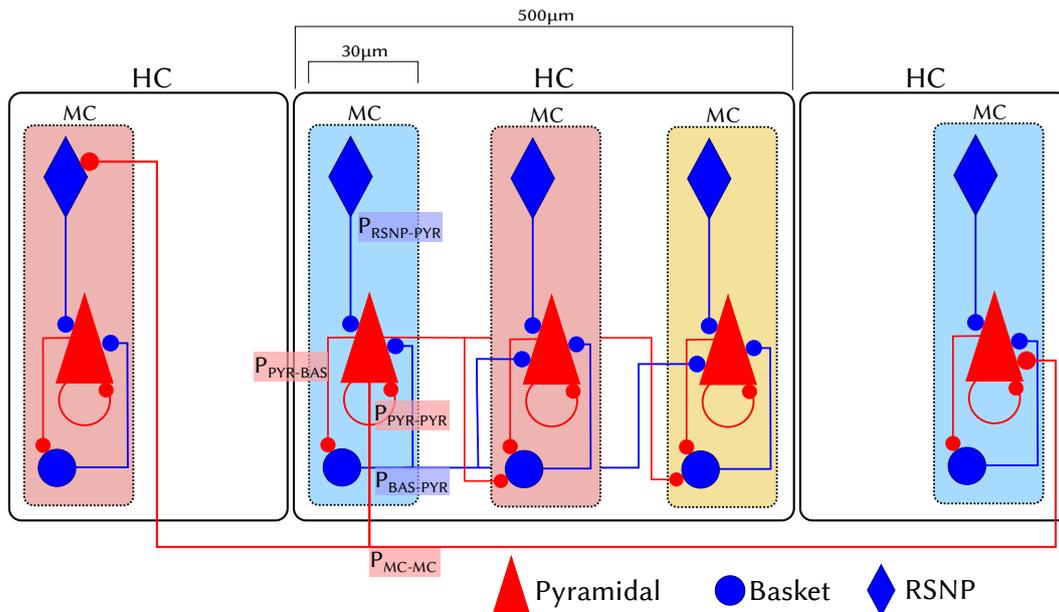


Figure 2.2: Layer 2/3 Attractor-Memory schematic. Shown are all connections emerging from a single MC. Excitatory connections are displayed in *red* while inhibitory connections are shown in *blue*. Patterns are represented by corresponding coloring of the MCs. Note that all connections between MCs are excitatory. Depending on whether two MCs share a common pattern or not, they target each others pyramidal or RSNP populations.

in figure 2.2. In each MC the pyramidal cells are interconnected with a probability of 25 %, while connections to and from the single basket cell occur with a probability of 70 %. The RSNP cells project onto the pyramidal cells with a probability of 70 % as well.

Each basket cell receives input from pyramidal cell populations in MCs of close proximity. The original model employed a single 8 basket cell population per HC receiving local input from all pyramidal cells in the HC. In the new setup this was translated to each pyramidal population projecting onto the basket cells in the eight closest MCs with a probability of 70 % – this becomes important when scaling the network to larger sizes (see 2.1.5 *Network Scaling*). Each basket cell in return projects onto *all* pyramidal cells within the HC with a probability of 70 %.

This creates a soft winner-take-all (WTA) situation within the HC, as an active MC will inhibit all other MCs via the basket cells. The change from *Lundqvist et al. [2006]* to have a single basket cell per MC opposed to a fixed number of basket cells per HC was made in order to naturally scale the numbers of pyramidal, RSNP and basket cells when increasing the number of MCs (see 2.1.5 *Network Scaling*). Long range connections between MCs in different HCs are governed by so-called patterns (see below).

2.1.2 Patterns

The logical building blocks of the network are patterns. Each attractor of the network is associated with exactly one pattern that defines which MCs make up the corresponding attractor. They are comprised of precisely one MC from every HC. Two patterns are said to be orthogonal if no MC in either pattern partakes in both. Unless otherwise noted all networks are comprised of orthogonal patterns where all MCs with the same index participate in the same pattern.

Patterns govern the long-range connections between MCs in *different* HCs. If two MCs participate in the same pattern, their pyramidal populations will be interconnected with a probability of 70 % (the MCs excite each other). Whereas, if the two MCs do not share any pattern with each other there will be excitatory connections from the pyramidal cells of the one MC to the RSNP cells of the other and vice versa (the MCs inhibit each other).

It is important to note the binary nature of these connections: All MCs from different HCs are interconnected, either in a purely excitatory (same attractor/pattern) or a purely inhibitory (different attractor/pattern) manner. Furthermore, all excitatory and all inhibitory connections are of the same density. As an example, it does not matter if two MCs are participating in one or three patterns together, the resulting excitatory connection will be the same.

This creates a “strong” WTA situation between all attractors in the network, as an active attractor will completely suppress all others, while the participating MCs excite each other.

2.1.3 UP-states

UP-states are a result of the intra-attractor stimulation as well as the cross-attractor inhibition. An attractor (meaning all its constituent pyramidal cells) is in an UP-state when the pyramidal cells in each MC participating in an attractor pattern are firing at a significantly elevated rate while all other attractors are suppressed. Local UP-states have been observed *in vivo* (Cossart *et al.* [2003], Shu *et al.* [2003]). About 0.5 % of the recorded pyramidal cells enter an UP-state simultaneously which then lasts several hundred ms (Lundqvist *et al.* [2006]). Due to neuronal adaptation as well as synaptic depression the dominating attractor’s activity will eventually decline, making way for other attractors to enter an UP-state.

2.1.4 Modeling

Neurons

Originally, each neuron was modeled using a Hodgkin-Huxley type model with six compartments (Lundqvist *et al.* [2006]), including soma, initial segment, basal dendrite compartment and three consecutive apical dendrite compartments. In order to make an emulation on the FACETS neuro-morphic systems possible, the hardware neurons needed to be fitted accordingly.

The FACETS Wafer-Scale Hardware implements the so called Adaptive Exponential integrate-and-fire model (AdEx model) (see Brette and Gerstner [2005]). It is described by the following differential

equations which are formulated in terms of the parameters defined in *PyNN* (see 2.3 *Implementation*).

$$\tau_m \frac{dV}{dt} = -(V - E_L) + \Delta_T \exp\left(\frac{V - V_T}{\Delta_T}\right) - \frac{\tau_m}{C_m} \cdot w + \frac{\tau_m}{C_m} \cdot I \quad (2.1)$$

$$\tau_w \frac{dw}{dt} = a(V - E_L) - w \quad (2.2)$$

Here, V is the membrane potential, w the adaptation variable, τ_m the membrane time constant, τ_w the adaptation time constant, E_L the leak reversal potential, C_m the membrane capacitance, V_T the threshold voltage, I the injection current, Δ_T the slope factor and a the adaptation coupling parameter. Whenever the membrane potential reaches a set value V_{spike} , a spike is said to occur and both the membrane potential V as well as the adaptation parameter are updated as follows:

$$\begin{aligned} V &\longrightarrow V_r \\ w &\longrightarrow w + b \end{aligned} \quad (2.3)$$

where V_r is the reset value for the membrane potential and b is the spike-triggered adaptation constant.

The AdEx model was used to describe the behaviour of both pyramidal and RSNP cells. Since in the original model basket cells are non-adapting, a Leaky integrate-and-fire (LIF) model was used. This corresponds to equation (2.1) without the adaptation parameter w and the exponential term and can easily be accommodated by the hardware neurons as well. The neuron parameters used in the default network are found in table A.3.

Synapses

The synaptic current I from equation (2.1) is generated via

$$I(t) = \sum_i^{N_{\text{spikes}}} g_w \exp\left(-\frac{t - t_i}{\tau_{\text{syn}}}\right) \Theta(t - t_i) (V_{\text{rev}} - V(t)) \quad (2.4)$$

where t_i denote the single presynaptic spike times, g the base synaptic conductance (or weight), Θ the Heaviside step function and V_{rev} the reversal potential of the respective synapse type (excitatory or inhibitory).

The synapses between all neurons are modeled as being depressing, according to the Short Term Plasticity (STP) mechanism proposed by Tsodyks and Markram (*Tsodyks and Markram* [1997]), which introduces the concept of synaptic resources. Whenever a presynaptic spike arrives only a portion U of the active fraction of resources R is accessible. Each spike activates a portion U of R , which therefore instantly decreases by a factor of $(1 - U)$ and then recovers slowly along an exponential with time constant τ_{rec} . When the $(n + 1)$ th spike arrives, R and the effective synaptic weight g_e are determined as

$$g_{n+1} = g_{w,\text{max}} R_{n+1} U \quad (2.5)$$

$$R_{n+1} = 1 - (1 - R_n(1 - U)) \exp\left(-\frac{\Delta t_{n,n+1}}{\tau_{\text{rec}}}\right) \quad (2.6)$$

where $\Delta t_{n,n+1}$ denotes the time interval between the n th and $(n + 1)$ th presynaptic spike. The synaptic weights used in the default network are found in table A.3.

Delays

Each connection was set to have a constant synaptic delay of 0.6 ms which is applied to all connections within the same MC. Additionally, axonal delays were realized for connections between different MCs by taking their spatial distance and dividing it by an average axonal velocity¹ of $0.2 \text{ m/s} = 200 \text{ }\mu\text{m/ms}$.

2.1.5 Network Scaling

The network allows for scaling in several natural ways which are employed throughout this thesis. The size of attractors can be varied evenly by changing the number of HCs, while – in case of orthogonal patterns (see *2.1.2 Patterns*) – the number of attractors can be changed by increasing or decreasing the number of MCs. Furthermore, the number of cells in each MC is open to modification.

Without additional changes in connectivity, such modifications would radically alter the network dynamics, as the average number of afferent synapses per cell is changed. A straightforward way to compensate for this is by adjusting the connection probabilities in a way as to preserve the average fan-in per neuron, hence not altering single cell dynamics. Since the model parameters were fitted using a default network size of 9 HCs with 8 MCs each (see *3.1 Parameter Fit*), two general types of scaling have to be differentiated: enlarging as well as shrinking the network.

In the first case, adjusting the probabilities is straightforward, because if the population sizes are increased, the probabilities have to be decreased in order to keep the number of realized connections for each cell constant. Accordingly, in the latter case, the probabilities have to be increased, but since they may not be larger than 1 (corresponding to every possible connection being realized), further compensation has to be performed via weight increases of the corresponding connection.

An exception is the connectivity from and to basket cells. As the network is shrunk, the total number of basket cells per HC falls below 8. In order to maintain the level of inhibition when one MC participates in an UP-state, the density of the remaining basket to pyramidal connections has to be increased. This is different from the case when the network is enlarged: Since each pyramidal population projects only onto the closest 8 basket cells – and all basket cells in turn project onto all pyramidal cells – the level of inhibition during an UP-state is the same regardless of network size. Hence, no further tuning is required.

All scaling formulae can be found in table A.4.

2.1.6 Stimulus

The network receives two distinct types of stimulus, namely unspecific background noise and input from cortical layer 4.

¹Personal correspondence with Mikael Lundqvist

Noise

All pyramidal cells in the network receive excitatory unspecific input from various other areas and structures. This is realized via Poisson processes in which all spikes arrive by *pure chance* and independently from each other. It follows a Poisson distribution $X_{\Delta t}$

$$P(X_{\Delta t} = n) = \exp(-\lambda\Delta t) \frac{(\lambda\Delta t)^n}{n!} \quad (n, \lambda \geq 0) \quad (2.7)$$

which denotes the probability for receiving n spikes during time Δt . λ is the spike rate of the Poisson process, so that the expectation value of $X_{\Delta t}$ is $\lambda\Delta t$.

For each pyramidal cell in the network independent processes with $\lambda = 300$ Hz are used. The synaptic weights are roughly 10 % of regular inter-pyramidal connections. With all further input blocked, each pyramidal cell spikes with a rate of 3 Hz; if only the inter-pyramidal connections are blocked, the spike rate depends on the network size, but lies around 1 Hz and below (see figure 2.3). This is somewhat lower than the 2.1 Hz (inter-pyramidal connections blocked) and 3.5 Hz (all further input blocked) spike rates observed in *Lundqvist et al. [2006]* and might indicate a stronger inhibition in the new model.

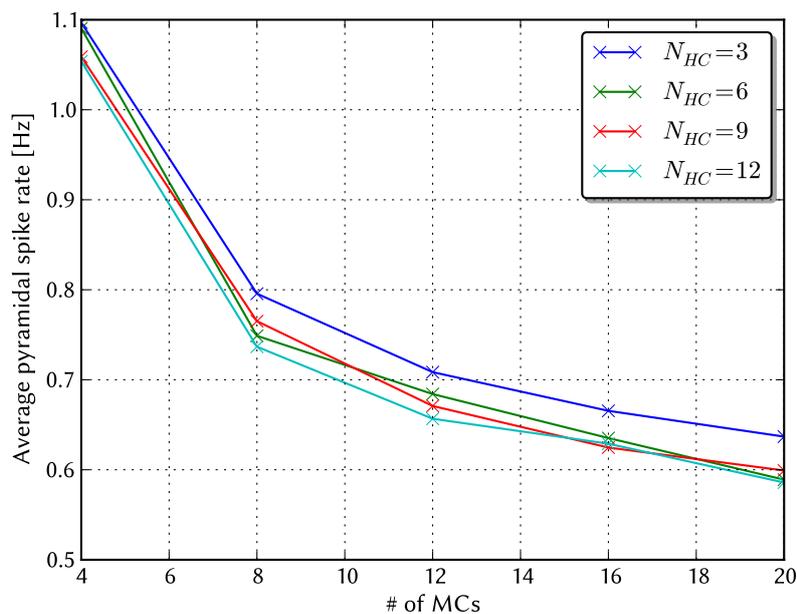


Figure 2.3: Average spike rate of a pyramidal cell in the network only due to noise. All other excitatory input to the pyramidal cells was blocked. Each network was simulated for 3000 ms.

Layer 4

Apart from random noise the pyramidal cells further receive input from pyramidal cells in cortical layer 4. The cells in layer 4 project strongly on layer 2/3, which this model covers. In the brain layer 2/3 itself provides feedforward connections to the layer 4 of the next higher cortical area as well as project onto deeper layers (*Miller [2003]*). The input intensity was calculated from the number of

cells in layer 4 likely to project onto layer 2/3, which was estimated to be around 30 with a rate of ~10 Hz and a connection density of 25 % (*Thomson et al. [2002]*).

Therefore a Poisson process with 75 Hz was used for each pyramidal cell receiving input. The weight is the same as for inter-pyramidal connections and static synapses were used. This was tested against actually realizing 7 to 8 sources per stimulated pyramidal cell with 10 Hz rate and depressing synapses; the difference is negligible since at 10 Hz inter-spike intervals are long enough to fully recover all synaptic resources.

2.2 Network Behavior Analysis

The investigated attractor-memory network has many interesting properties that first have to be extracted from the raw data. Therefore, the analysis module was substantially extended. The most important improvements are discussed in this section.

The basis for all analyses is the data recorded during simulation. It is comprised of spike times from *all* cells, to 0.1 ms accuracy, as well as voltage data (again in 0.1 ms intervals) from the RSNP cells and a subsample of all pyramidal cells, as it was not feasible to record all voltages from all cells due to memory constraints.

2.2.1 Data Preprocessing

The recorded data needs to be preprocessed before the analysis can proceed. The recorded voltages from single cells contain gamma rhythm-like oscillations, that have been shown to correlate with the delay activity in a memory task (*Tallon-Baudry et al. [1998]*) and are also investigated in *Lundqvist et al. [2006]*. This is of lesser concern if the average membrane potential for the entire attractor is to be inferred. Simply averaging over all recorded participating cells is not enough, as this does not eliminate the short time oscillations (see figure 2.4b); the data needs to be filtered.

Furthermore, in order to compute actual spike rates from the list of spike times, those have to be filtered as well.

If the raw signal recorded is denoted by $s(t)$, we have

$$s(t) = \begin{cases} \frac{1}{N_{\text{records}}} \sum_i v_i(t) & \text{for voltage time courses} \\ \sum_j \delta(t - t_j) & \text{for spike trains} \end{cases} \quad (2.8)$$

where N_{records} denotes the number of recorded cells, $v_i(t)$ are the actual recorded voltage traces, N_{spikes} denotes the number of spikes, t_j the single spike times, and δ is the Dirac delta function satisfying $\int f(x)\delta(x - x_0)dx = f(x_0)$.

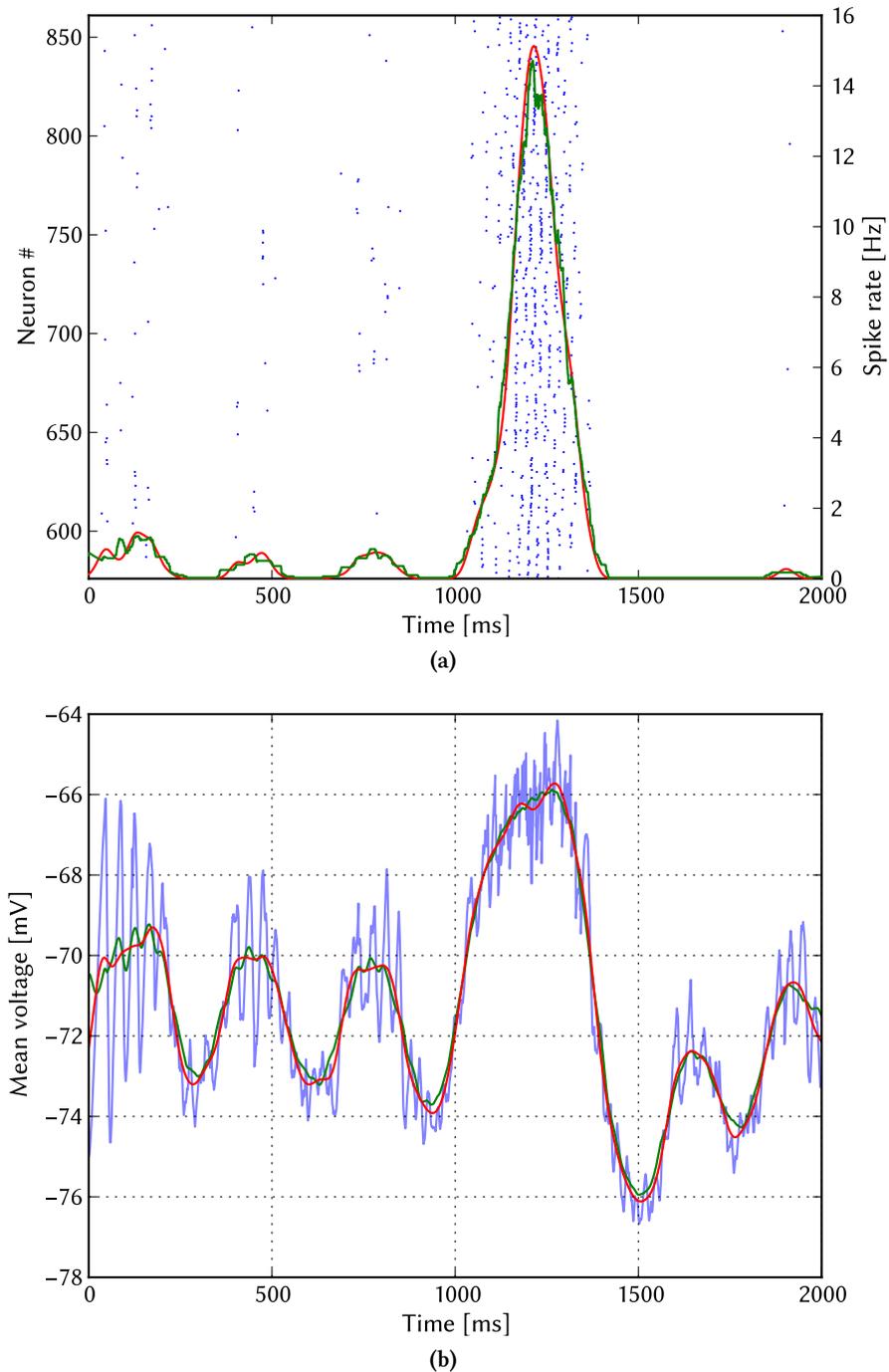


Figure 2.4: Comparison between a box filter ($\Delta_b = 100$ ms) and a Gauss filter ($\sigma = 20$ ms $\rightarrow \Delta_g = 160$ ms) on data from a single attractor. The network size was 9 HCs with 8 MCs each: (a) The spike times (blue) and their convolution with box (green) as well as Gauss filter (red). (b) Average recorded voltage (blue) from 18 pyramidal cells (2 per MC) according to (2.8) and its convolution with both box (green) and Gauss filter (red). The large peak at around 1000–1500 ms corresponds to an active UP-state (see raster plot above). The oscillations still present in the signal correspond to the competition times between UP-states of single attractors. Small peaks appear when all attractors compete to be the next one to enter an UP-state, whereas the valleys correspond to the suppression of another attractor in an UP-state, as can be seen by the fact that a valley is missing when the attractor observed here enters its UP-state.

Box Filter

A first implementation made use of a box filter

$$\tilde{s}(t) = \int \frac{1}{b} \Theta \left(\frac{\Delta_b}{2} - |t - t'| \right) s(t') dt' \quad (2.9)$$

where $s(t)$ is the filtered signal, Θ is the Heaviside function and Δ_b is the width of the box filter. This relatively simple approach allows a very fast implementation: The filter is slid along the data while in each step one data point is added and one is dropped, amounting to a $\mathcal{O}(t_s)$ run time, where t_s denotes the total biological simulation time. Its main drawback lies in the fact that \tilde{s} is a discontinuous function of time due to the discontinuous nature of the (filter) kernel itself.

Gauss Filter

A possible approach to counter this problem is to convolve the original signal with a Gaussian

$$\tilde{s}(t) = \int \frac{1}{\sqrt{2\pi}\sigma} \exp \left(-\frac{1}{2\sigma^2} (t - t')^2 \right) s(t') dt' \quad (2.10)$$

where σ is the width of the normal distribution. The resulting signal is smoothed without any discontinuities (see figure 2.4 for a comparison). A value of 20 ms was used for σ throughout this thesis. Since more than 99.99 % of a normal distribution is contained within a $\pm 4\sigma$ interval around the mean, all values with $\Delta T = |t - t'| > 4\sigma$ were omitted when computing the convolved signal \tilde{s} at time t . The width of the filter was hence $\Delta_g = 2 \cdot 4\sigma$. This is important because, as opposed to the simpler box filter, the computational cost for Gauss filtering is $\mathcal{O}(t_s \cdot \Delta_g)$.

2.2.2 Phase Space Trajectory Projection Plots

A very efficient way of visualizing the dynamics of the network is by virtue of PSTP plots. The dynamics of the network can be viewed as a trajectory in an abstract feature space, spanned by all the attractors. Suitable features are the mean voltage or mean spike rate of each attractor. Since in general, there will be more than three distinct attractors, the feature space will have more than three dimensions and is therefore not visualizable in a straightforward way.

However, the complete trajectory can be projected onto a hyper-plane perpendicular to the main diagonal (see figure 2.5). Obviously, a lot of information is lost in the process. Still, the network can clearly be observed entering distinct activity states (see figure 2.6).

Additionally, the velocity v of the trajectory is visualized by means of line color and thickness. Both are scaled exponentially so that the color darkens and the size increases as the velocity decreases.

$$\text{thickness, color} \propto \exp \left(-\frac{v}{v_{\max}} \right) + \text{offset} \quad (2.11)$$

where several scaling factors have been omitted which were set to give a good visual representation of the velocity.

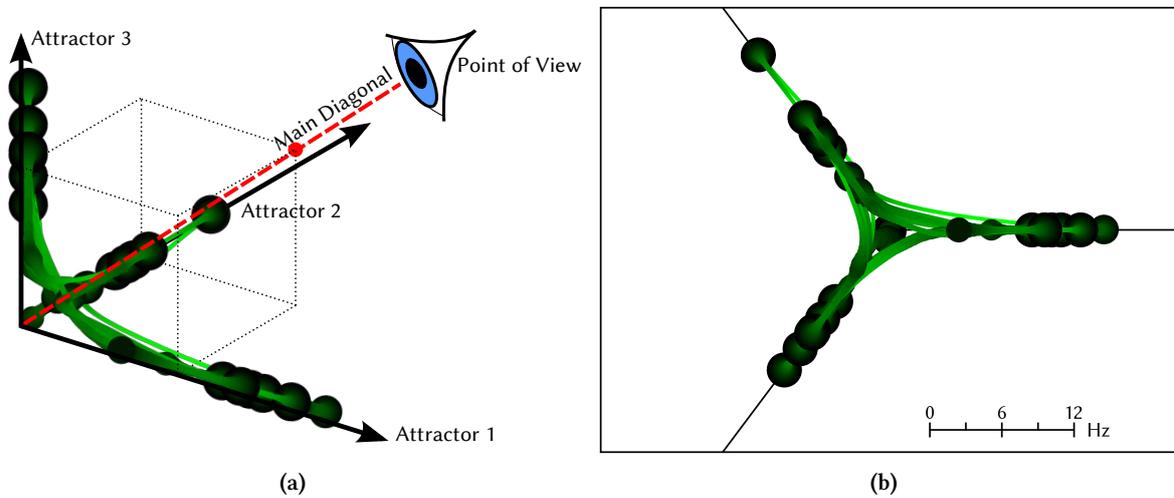


Figure 2.5: Illustration as to how PSTP plots are generated: (a) The trajectory in a high-dimensional (here three-dimensional, taken from a network with 9 HCs and 3 MCs) feature space, spanned by the attractors of the network, is projected onto a hyper-plane perpendicular to the main diagonal. The projection is the regular PSTP plot as seen in (b).

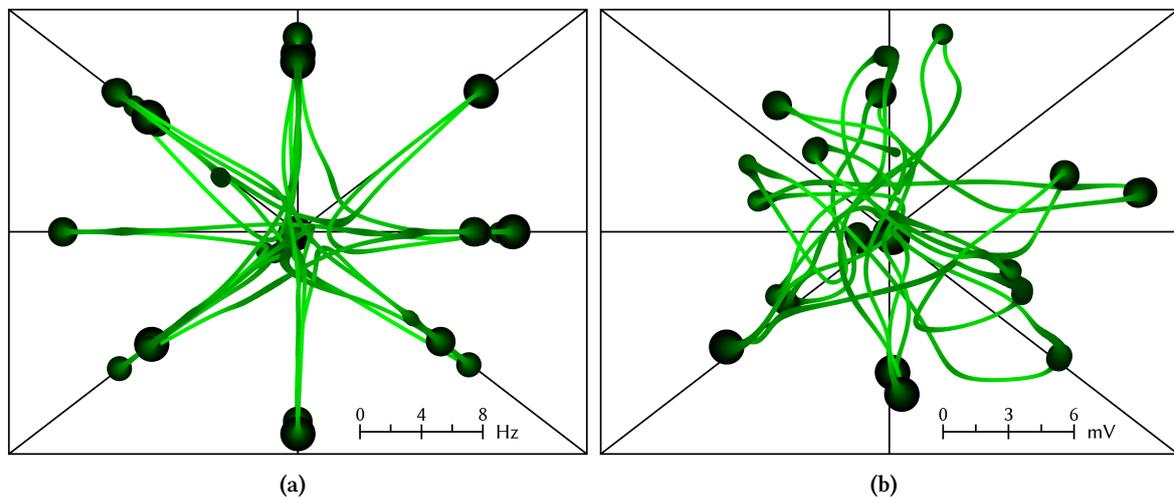


Figure 2.6: Phase Space Trajectory Projection plots of a network with default size (9 HCs with 8 MCs each). (a) Mean spike rate trajectory: The distinct UP-states of the network are observed. (b) Mean voltage trajectory: In contrast to the mean spike rate plot the trajectory is not as aligned with the attractors. This is due to the fact that in case of an UP-state the other attractors' spike rates are close to zero, while their membrane potentials still vary due to constant inhibition through the active attractor as well as excitation through background stimulus. Note that the blobs corresponding to a slow phase velocity are usually far from the center, meaning that not many attractors have a high membrane potential at the same time during an UP-state.

2.2.3 UP-state Detection

One crucial element of the analysis is the detection of UP-states from which various other properties such as dwell times, competition times as well as average spike-rates in UP- and DOWN-states are determined.

Voltage-based

The first implementation was based on recorded voltages. For each attractor a threshold value $V_{th} = \frac{1}{2}(V_{max} + V_{min})$ was determined, where V_{max} and V_{min} were taken from the mean voltage traces recorded from a fixed number of pyramidal cells in each MC participating in the pattern. Whenever the average voltage would go above the threshold, the attractor was said to be in an UP-state. For small, well-defined networks, in which all attractors were active at least once and competition times between attractors were small, this was an acceptable measure.

However, for more general networks investigated in this thesis it had several shortcomings. Firstly, there was no notion that only one attractor could be in an UP-state at any given time since each attractor was evaluated separately. This could lead to reduced competition times and even overlapping attractors. Secondly, the detection was dependent on each attractor being active at least once. Otherwise V_{max} and V_{min} were determined by random membrane oscillations, yielding false-positive detections of extremely short UP-states. The third and most severe drawback was that it relied on recorded voltage traces which – because of memory limitations in software – can only be efficiently recorded from a subset of all pyramidal cells² and are also only restrictively accessible on the FACETS hardware (2 out of 512 neurons can be recorded per HICANN, see *Millner et al. [2010]*). Hence the goal was to find a reliable method of detecting UP-states that could work on spike rates alone if necessary.

Score-based

A new, score-based method was developed which evaluated certain features of both voltage and spike rate data preprocessed by the method to produce the star plots (see figure 2.6 and section 2.2.2 *Phase Space Trajectory Projection Plots*). The features used were angle and radius of the phase space projection and its *angular* velocity in both mean rate and mean voltage phase space, as well as the mean spike rates of the RSNP cells in each attractor. Each set of data was optional, the method could also be applied to either voltage or spike rates alone.

For each time step a candidate attractor was generated (corresponding to the attractor with the highest voltage or – if available – the highest spike rate at that time), for which it was then determined whether or not it actually was in an UP-state. Then certain criteria were checked:

- Is the distance of the position in the phase space projection sufficiently far away from the origin? Otherwise this would indicate competition between several attractors.
- Does the angle correspond to the candidate within a certain margin of error?
- Does the angular velocity not exceed a certain value? Note that by inspecting the angular velocity, rate changes within an active attractor are ignored, because those are aligned with the attractor and hence correspond to changes in the radius of the PSTP rather than its angle.

²The subset was comprised of 2 randomly chosen pyramidal cells per MC – in larger networks the sample size had to be reduced to 1.

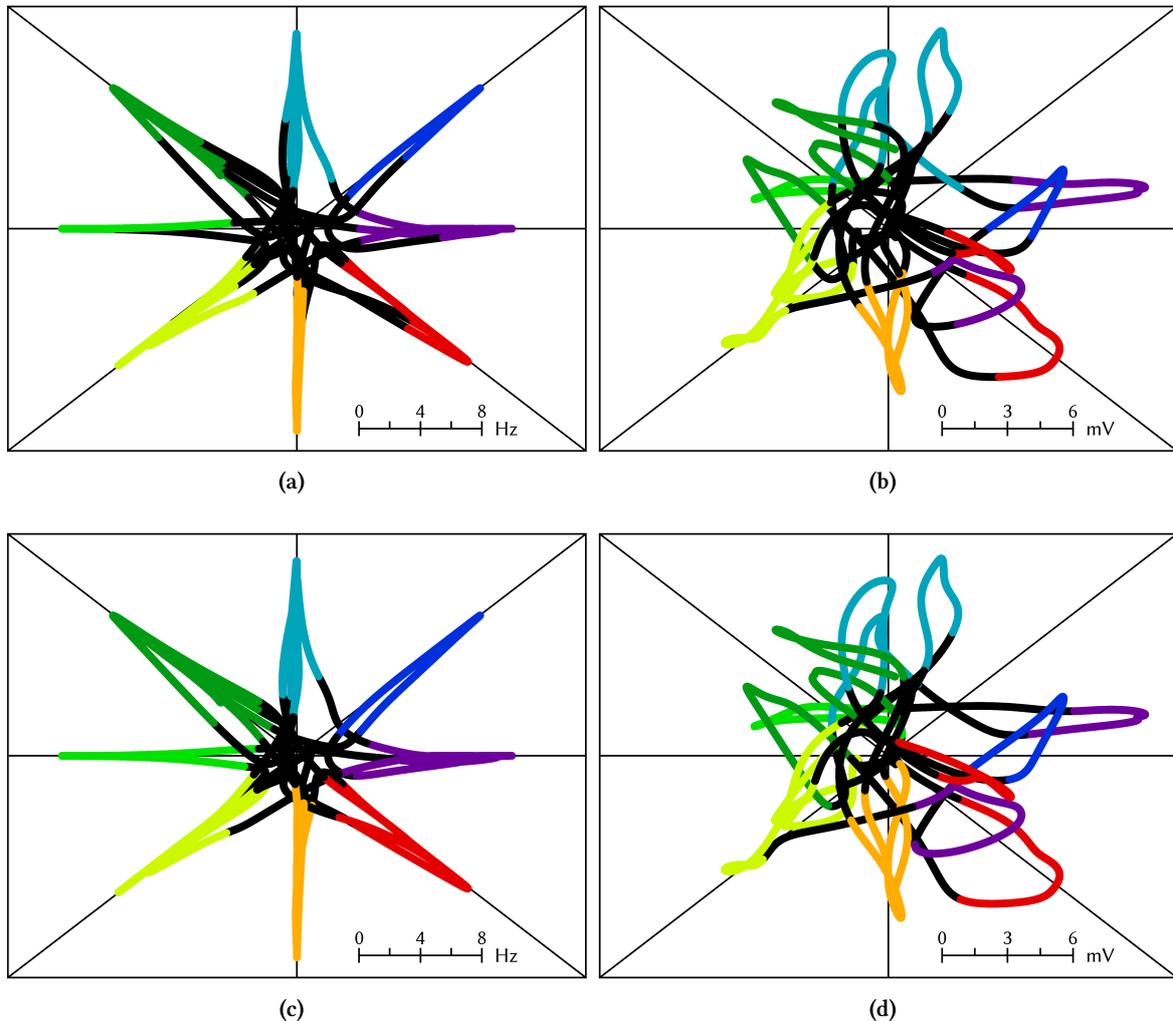


Figure 2.7: Comparison between score and spike based UP-state detection: PSTP plots from figure 2.6 without velocity information. Instead, colors denote which attractor was detected to be in an UP-state.

Example of score-based UP-state detection on a default sized network (9 HCs by 8 MCs) : (a) Spike rate phase space, (b) average voltage phase space. Violating angular alignment in voltage phase space is weighted less than in spike rate phase space. Furthermore each successful UP-state is more likely to occur far from the origin. Overall this leads to shorter detected UP-states and the problems mentioned in the text.

The same network is analyzed via the spike-based method described in 2.2.3 *Spike-based*: (c) spike rate phase space, (d) average voltage phase space. With much fewer adjustable parameters, the constraints applied to PSTPs are rather similar.

- Does the candidate attractor have the lowest spike rate of the RSNP cells? Since only those RSNP cells receive input which are not participating in an UP-state, the RSNP spike rate of the active attractor should be lowest.

Each criterion that was not met was penalized with an individually adjustable score. This allowed a weighting of features based on their importance: An example would be the angular alignment of the spike rate PSTP plot (see figure 2.6a) versus the voltage PSTP plot (see figure 2.6b). If the overall score was lower than an adjustable threshold, the candidate was said to be in an UP-state, otherwise the complete network was in a state of competition.

While this detection mechanism did allow reliable UP-state detection on spike rates only if desired, finding adequate values for each adjustable parameter proved to be both time consuming and rather arbitrary. Furthermore, the score-based algorithm had the tendency to detect very short DOWN-states between prolonged UP-states of the same attractor. This became rather hard to eliminate, which is why this method was abandoned once the following method was conceived.

Spike-based

The final method for detecting UP-states – and the one which is employed throughout this thesis – is based on the rather trivial observation that the mean spike rate of an attractor during an UP-state is much higher than the spike rate in all remaining patterns in their corresponding DOWN-states, whereas – in times of competition – two or more attractors have elevated but rather similar spike rates. A measure which quantifies this relationship is the standard deviation σ of all mean spike rates *at a given time* t .

$$\sigma(t) = \frac{1}{N_{\text{Pat}} - 1} \sqrt{\sum_{i=1}^{N_{\text{Pat}}} (r_i(t) - \bar{r}(t))^2} \quad \text{with} \quad \bar{r}(t) = \frac{1}{N_{\text{Pat}}} \sum_{i=1}^{N_{\text{Pat}}} r_i(t) \quad (2.12)$$

where N_{Pat} denotes the number of patterns and $r_i(t)$ is the rate of pattern i at time t . The attractor with index i is then said to be in an UP-state at time t if the following relation holds true

$$r_i(t) > c \cdot \sigma(t) > r_{1..N_{\text{Pat}} \setminus i}(t) \quad (2.13)$$

Where c is a numerical constant which is set to 1 for orthogonal patterns, but has to be increased in case the spike rates correlate too much in case of non-orthogonal patterns (see section 3.4 *Non-Orthogonal Patterns*). An illustration of this principle is outlined in figure 2.8.

This method of detection has the advantage that it is based exclusively on spike trains, has a clear notion of there being – at most – one UP-state at any given time and is a completely local measure, meaning that a very large value somewhere on the time axis cannot bias the detection at all other times. Both previous methods had this disadvantage in that they were dependent on extremal values extracted from the *complete* data itself. As a test, model networks which have been designed to have no UP-states at all are analyzed correctly, as long as the total number of (inactive) attractors is large enough. In too small networks with randomly spiking neurons, it might happen by chance that all but one spike rate is below the more or less constant standard deviation. However, this probability decreases exponentially as the number of attractors increases and is effectively gone once the network size reaches 6 attractors (see figure 2.9). Furthermore, the falsely detected UP-states are very short and can thus easily be filtered out if so desired.

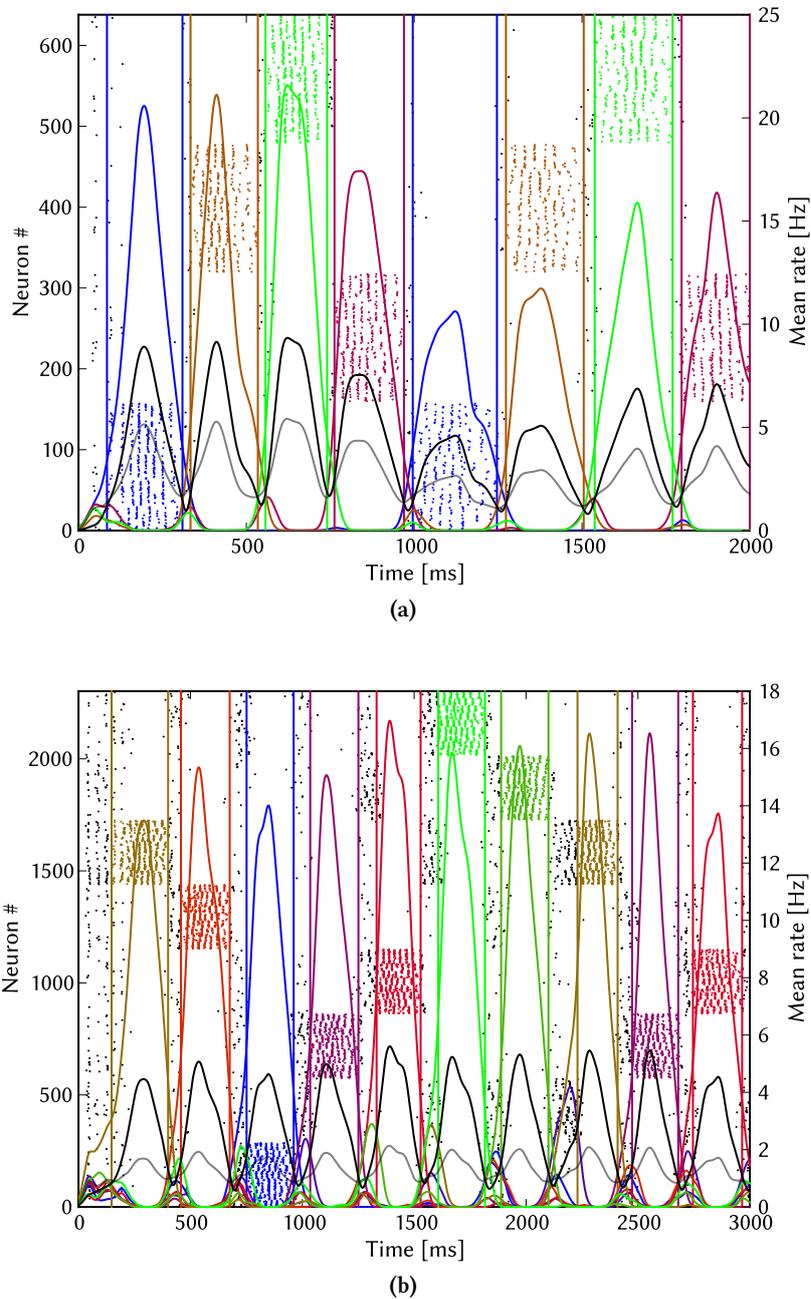


Figure 2.8: Illustrating the working principle of the UP-state detector for a small network (a) and a default sized network (b). The small network is comprised of 5 HCs with 4 MCs each, while the default network contains 9 HCs, each with 8 MCs. Shown are the spiking neurons and their corresponding mean spike rate, color coded by pattern. The standard deviation σ (black) and average spike rate of the whole network (grey) are plotted as well. It can be clearly seen that during an UP-state, the active attractor's spike rate is separated from all other attractors' spike rates by the standard deviation σ . While the mean rate $\bar{r}(t)$ (gray) has the same property, it lacks variability and is therefore less sensitive to times of competition. UP states with different spike rates are detected as well.

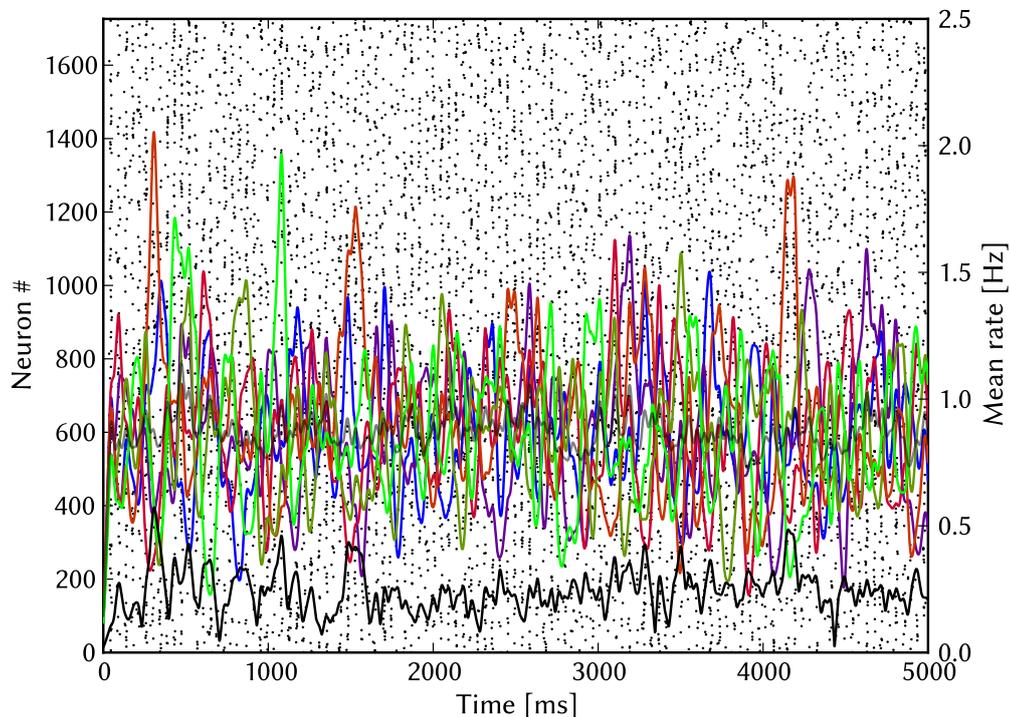


Figure 2.9: Spike-based classification on a network with 9 HCs by 6 MCs where the excitatory connections between pyramidal cells have been severed. The important thing to note in this plot is the fact that since all spike rates are rather similar the standard deviation (*black*) is small. Hence no UP-states are being detected, which is the desired outcome.

2.3 Implementation

The complete model is implemented in PyNN (*Davison et al. [2008]*), which is a back-end-agnostic Application Programming Interface (API). It provides a library of standard neuron and synapse models and allows high-level, object-oriented modeling of neuronal networks in the programming language *Python*. In principle, the simulator can be switched by changing a single line of code. It therefore allows simulation of the same neuron models and network architectures with different backends. These backends include several software simulators as well as the virtual hardware (see for example *Brüderle et al. [2011]*).

The actual backend used in this thesis is *NEURON* (see *Hines and Carnevale [2006]*), a simulation environment for both single neurons and networks written in *C*, which was primarily developed by Michael Hines, John W. Moore, and Ted Carnevale at Yale and Duke.

All simulations were conducted with an *Intel Core i5 750* as well as an *AMD Phenom II X4 965* CPU. Both machines were equipped with 8 GB of main memory.

With PyNN as basis it was a straightforward decision to write analysis routines in *Python*, while – most prominently – using *NumPy* (*Oliphant [2006]*) and *Matplotlib* (*Hunter [2007]*).

Model Implementation

An early version of the model with basic functionality was provided prior to this thesis with a somewhat dissimilar network architecture as well as no way of defining patterns and stimulus to the network without changing the source code, which has since been updated to the present network structure. Also, among other minor things, the stimulus and pattern modules discussed below were designed and implemented, greatly enhancing the model's feature range.

Stimulus Module

An essential part of this thesis was the development of the pluggable stimulus module with which layer 4 input becomes possible. In the earlier implementation of the model only background stimulus was present. Now stimulus to single or logical groups of pyramidal cells is easily configurable without changes to the source code itself. Random sampling is supported as well – the user simply specifies a number of HCs/MCs/pyramidal cells to stimulate which causes a sample of the specified size to be drawn randomly for the network/each HCs/each MCs. Furthermore, MCs in a single HC can be stimulated based on certain properties (see 3.5 *Retinotopic Pattern Completion*).

It is configured via a pluggable configuration file which parses several parameters from the network parameter configuration file (e.g. number of HCs, MCs or pyramidal cells). This allows one input configuration to be used with several network layouts.

The complete input – or only the input different from background noise (see 2.1.6) – to the network can be recorded. Since the number of spike sources for each individual neuron can differ (depending on whether the neuron receives additional layer 4 input or not), a lookup table is saved as well so that each spike source can be matched to the pyramidal cell it was targeting during simulation.

Pattern Module

Another part of this thesis was the implementation of the pluggable pattern module with which the network structure can be configured freely in terms of which MCs participate in which pattern (used especially in section 3.4). The only reasonable limitation made was that each pattern consists of *one* MC out of *every* HC (as already outlined in 2.1.2 *Patterns*).

It follows the same design principle as the stimulus module in that patterns are defined in a separate configuration file in which placeholders may be used in order to make it compatible with several network sizes.

Furthermore, MCs participating in a pattern can be assigned to have a certain property represented by an integer. Stimulus configured with the stimulus module can then be set up to only target MCs within a HC which possess the specified property. With this setup it is possible to conduct more complex pattern recognition and completion experiments, such as the ones described in 3.5 *Retinotopic Pattern Completion*.

Analysis

The early version of the model was already accompanied by an analysis script, implemented in almost pure *Python*. Since then, the script has been greatly enhanced: Several more features are evaluated, plotted and saved, such as the preprocessed data, when which pattern is in an UP-state,

2.3 Implementation

dwell and competition times etc. All this data is then available for another series of scripts that extract and plot different features over a range of networks³.

Many features of the analysis – such as which UP-state detection to use or whether to plot several features for each pattern as well as PSTP plots – are configurable on a per simulation basis. This allows batch simulation and analysis of large numbers of networks with different parameters⁴. All data plots throughout this thesis were generated this way.

Furthermore, by making use of the array-wide operations offered in *NumPy*, the performance of all analysis scripts could be greatly enhanced (thereby decreasing run time).

Problems

The great flexibility of *Python* comes at a performance price, and so several steps had to be taken in order to achieve acceptable performance for both simulation and analysis of the model. Whenever possible, array-operations were carried out on *NumPy* arrays which are implemented in low level languages such as *C* or *Fortran*. More complex computations on even larger arrays of data (preprocessing the data, see 2.2.1 *Gauss Filter*) were implemented in *C*-code and plugged into the analysis script.

Another issue was the fact that the originally used *NeuroTools*⁵ module loaded all spike trains and voltage traces at once, even though that data was only needed once and in serial order for preprocessing. Especially for voltage data this led to huge amounts of memory being wasted, up to the point, that the simulation server was brought to a near halt, desperately swapping data in and out of the main memory. Therefore, a custom implementation of needed functionality from the `NeuroTools.signals.SpikeTrain` and `NeuroTools.signals.AnalogSignal` classes was implemented, thereby greatly reducing the needed memory both during analysis and when saving the simulation data after a successful run. Furthermore, all spike and voltage data is saved using custom implementations and *NumPy* routines, in order to emulate a *NeuroTools*-written file without its memory overhead.

³these scripts are, for example, used in 3.2.2 *Pattern Completion*, 3.2.3 *Pattern Rivalry* and 3.3 *Hardware Imperfections*

⁴e.g. with/without recorded voltages, several plots disabled which are not informative for very long biological times

⁵<http://neuralensemble.org/trac/NeuroTools>

3 Results

In this chapter the benchmark network model is systematically investigated. The list of analyzed features includes regular network dynamics and how they are affected by different network sizes. Furthermore, pattern completion and rivalry are investigated as well as how the network responds to three different kinds of hardware specific distortion mechanism. Due to the fact that most of the time in this thesis was spent developing the network model, the addition of non-orthogonal patterns as well as retinotopic pattern completion – found at the end of the chapter – are more intended to be a showcase of what the network is capable of performing, rather than an in-depth analysis.

The reader is also referred to the upcoming publication *Petrovici et al. [2011]* in which further analyses of the network model will be conducted.

3.1 Parameter Fit

Since there is no straightforward way to transform the parameters of the neuron model employed in the original network model, in which each neuron is modeled with six compartments, the parameters of the AdEx model (see *2.1.4 Neurons*) had to be fitted. Figure 1 in *Lundqvist et al. [2006]* was used for reference, the corresponding Post-Synaptic Potentials (PSPs) can be seen in figure 3.1.

All further parameters were either taken directly from *Lundqvist et al. [2006]* or adjusted as to reproduce roughly the same spiking rates in UP- and DOWN-states as in the original model (see *3.2.1 Background Input*).

Since in the original paper the network size was fixed to a default size of 9 HCs with 8 MCs, all parameters were fitted to that network size and then scaled for other network sizes (see *2.1.5 Network Scaling*).

One problem during the fitting process was the occurrence of extreme UP-states, in which the pyramidal cells would fire with roughly ten times the regular spike rate (which is observed in *3.2.1 Background Input*), greatly reducing their dwell time. The only way to avoid this phenomenon was to significantly increase the synaptic weight from RSNP and basket to pyramidal cells. This illustrates some of the difficulties encountered during parameter tweaking.

3.2 Orthogonal Patterns

First, general characteristics of a network with orthogonal patterns are evaluated in order to make a qualitative comparison between the new network model and the original possible. Orthogonal patterns are comprised of one MC out of each HC where no MC is in two patterns at once. Without loss of generality all MCs with the same index i are set to participate in the corresponding pattern i .

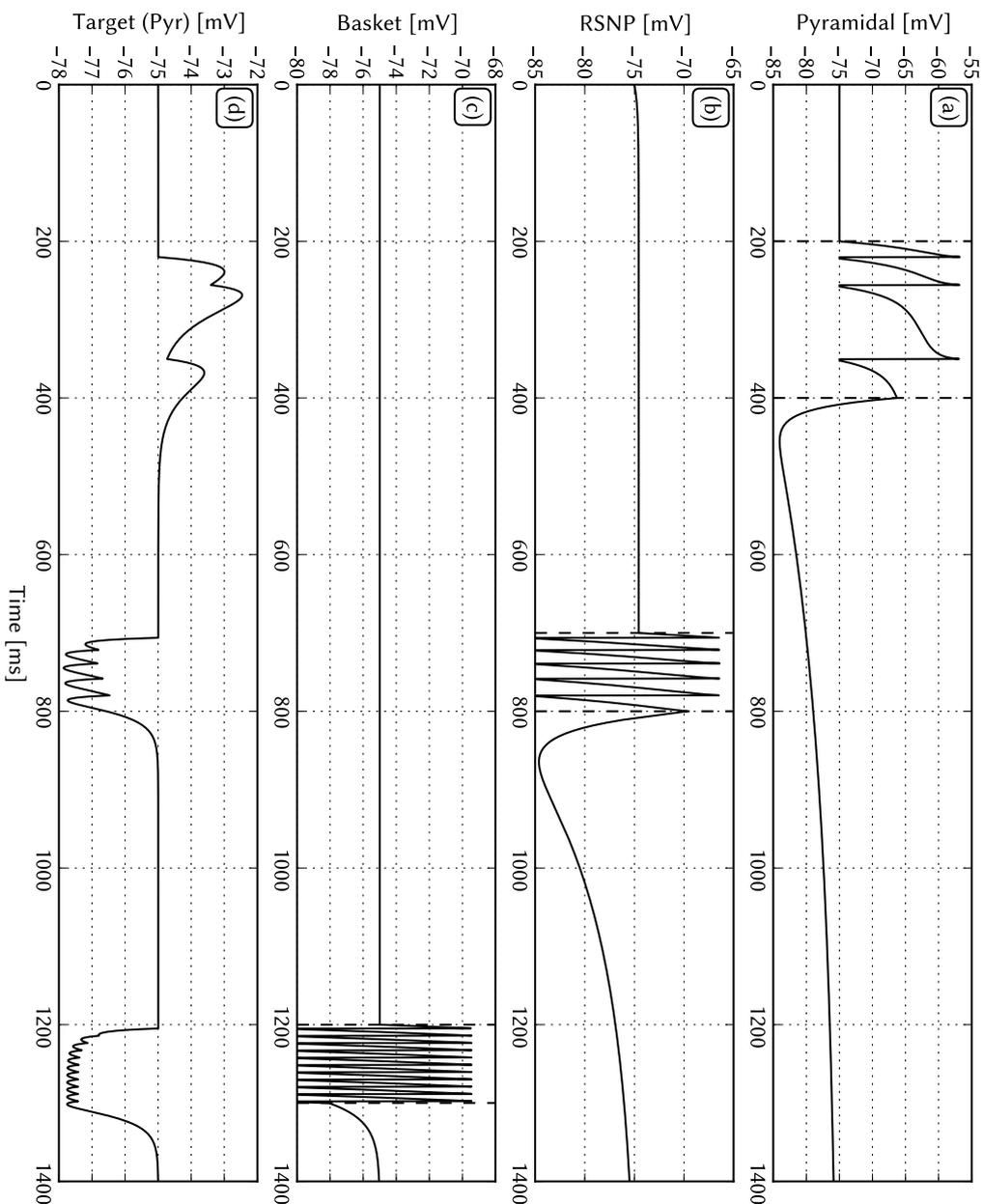


Figure 3.1: Fitting the parameters of the AdEx model to reproduce the PSPs observed in *Lundqvist et al.* [2006] (see figure 3.2). All three types of cells in the model are injected with currents over a period of time (*dashed black lines*):

- (a) pyramidal cell (0.1 nA, 200 ms),
- (b) RSNP cell (0.01 nA, 100 ms),
- (c) basket cell (0.01 nA, 100 ms),
- (d) PSP of a pyramidal cell due to presynaptic spikes arriving from (a)-(c). Both neuronal adaption (only in (a)) as well as synaptic depression are observed.

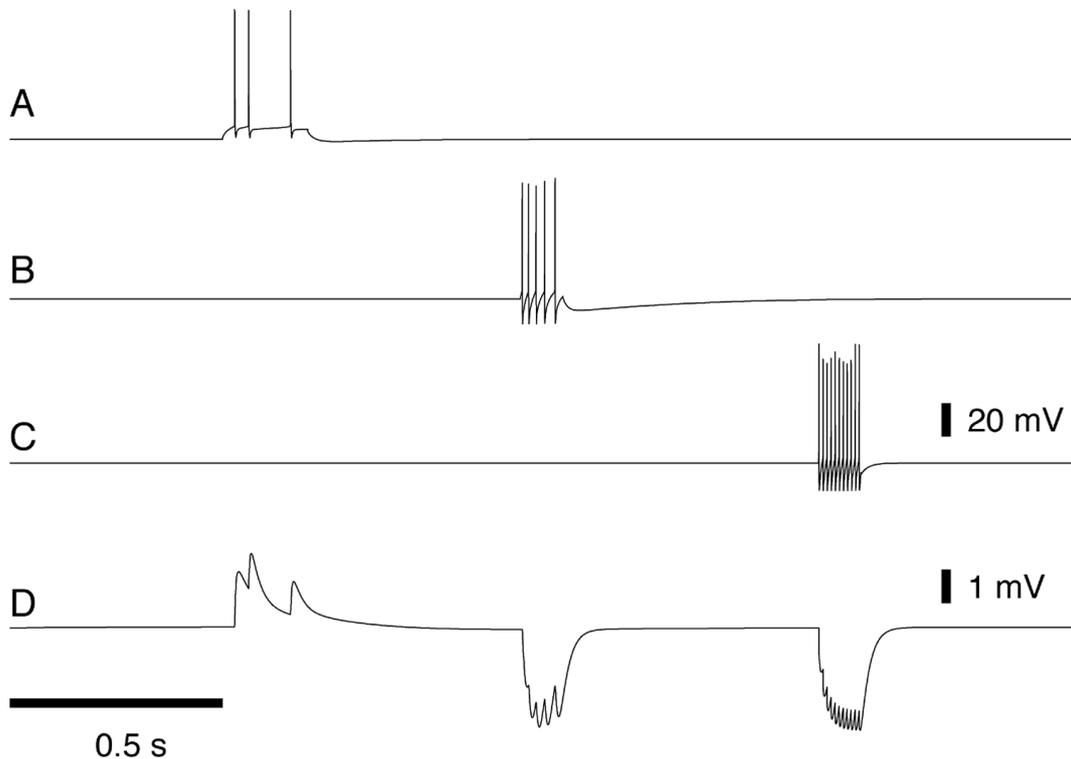


Figure 3.2: Original PSP plots in *Lundqvist et al.* [2006] (figure 1) that served as basis for fitting the parameters of the AdEx model employed in this thesis.

A to D in this figure correspond to (a) to (d) in figure 3.1.

3.2.1 Background Input

When stimulating the network with background noise only, it exhibits spontaneous UP-states (see figure 3.3). The dwell time of a single UP-state is roughly inversely proportional to the number of attractors. This is due to the increased competition between the different patterns, whereas increasing the number of HCs relative to the number of MCs stabilizes the UP-state since more MCs are participating in an active attractor (see figure 3.5).

Due to the fact that competition increases when the number of MCs is increased, competition times increase by a large amount in larger networks and spontaneous UP-states become increasingly rare (see figure 3.6). Whereas in the smallest network (6 HCs, 3 MCs) almost 90 % of the time one attractor was in an UP-state, the percentage decreases to 26 % (8 HCs, 20 MCs) and 17 % (18 HCs, 16 MCs) for larger networks. It is important to note that increasing the number of HCs or MCs *both* led to an increase in total competition time, albeit only the latter led to very long DOWN-states. This can be explained by the fact that either the relative strength of an individual competitor or the total number of competitors is increased.

Another feature to investigate is the mean spike rate during UP- and DOWN-states (see figure 3.7). In most cases the UP-state spike rates lie roughly within the range of 8–12 Hz. Due to the competitive nature of larger networks, less spontaneous UP-states occur in total. Since UP-states only last for about 200–250 ms on average, there is an inverse correlation between spike rate and dwell time: Since $\tau_w = 400$ ms for pyramidal cells (see table A.2), each pyramidal cell can only fire

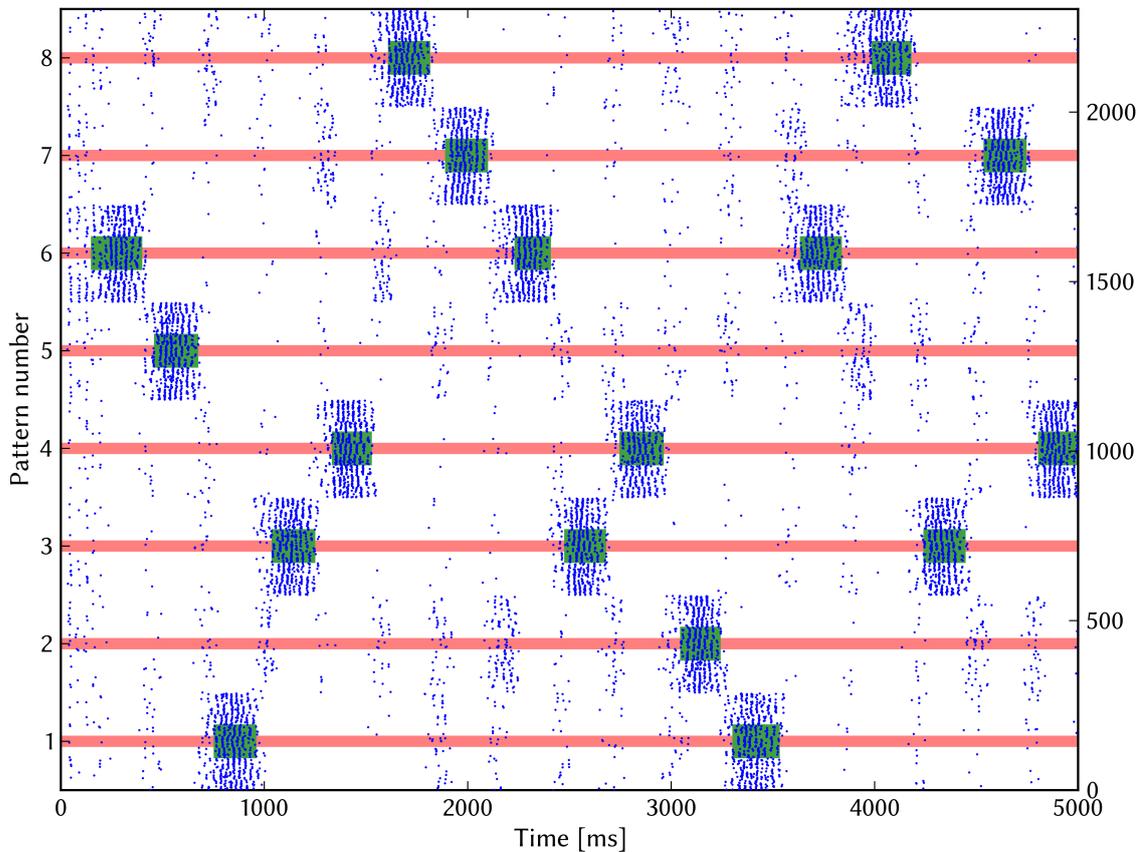


Figure 3.3: The default network: Pyramidal spikes for a default network consisting of 9 HCs with 8 MCs each stimulated by the noisy background only. The horizontal lines denote UP- (*green*) and DOWN-states (*red*) of the attractors and have been automatically generated by the analysis module. Spontaneous UP-states with short competition times in between are observed.

a certain number of spikes before the adaptation variable becomes too large for the pyramidal cell to spike again with the current input. Hence, the attractor dies out.

Compared to the original model, UP-states in the new model are only half as long (200–250 ms compared to 500–1000 ms). On the other hand, average spike rates in the range of 8–12 Hz are nearly double of what was observed for the original model (5.6 Hz). However, the peak spike rate of 14 Hz during an UP-state observed in the original model could be reproduced (see for example figure 2.4a). This suggests that the current parameters could be optimized to reduce average spike rate during an UP-state while increasing overall dwell times.

3.2.2 Pattern Completion

Pattern completion is a basic property of associative-memory networks. By only stimulating a subset of pyramidal cells participating in a pattern, the complete pattern is recalled, as the activity first spreads within the stimulated MCs, turning them dominant in their corresponding HCs. After that, the activity spreads further to other HCs – while the already dominating MCs stabilize each

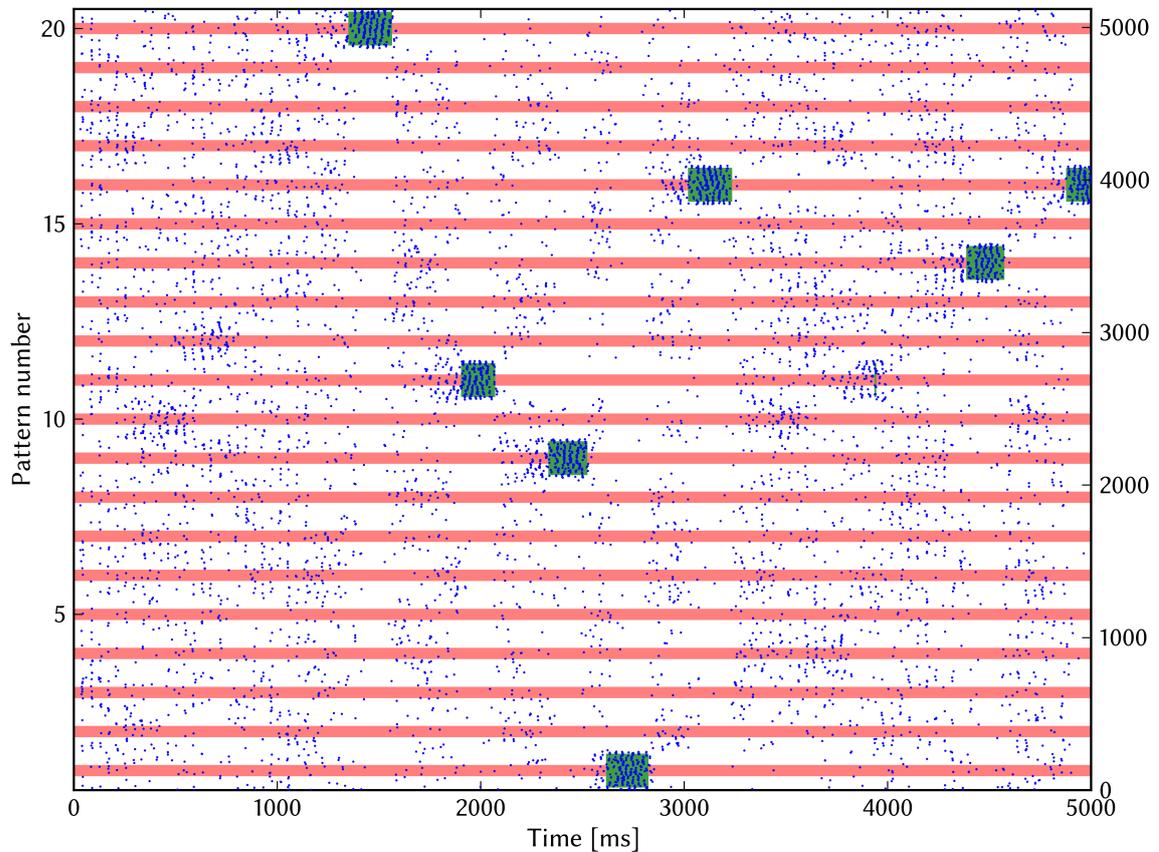


Figure 3.4: Pyramidal spikes for a larger network consisting of 8 HCs with 20 MCs each stimulated by the noisy background only. The horizontal lines denote UP- (*green*) and DOWN-states (*red*) of the attractors and have been automatically generated by the analysis module. Spontaneous UP-states are observed here as well, with much longer competition times (than in figure 3.3) in between.

other through mutual stimulation – activating the whole pattern while suppressing all others. The corresponding attractor hence enters an UP-state.

To verify the existence of pattern completion a series of different¹ networks was simulated. In order to reduce the occurrence of spontaneous UP-states – which would interfere with induced UP-states – competition within the network was increased. Each network’s size was changed to 8 HCs with 20 MCs with stimulated UP-states. Yet, the number of HCs was not increased in order to achieve reasonable simulation times on the employed hardware.

For each network, 7 out of the 20 total patterns were chosen at random. Every 700 ms one of those patterns was stimulated. To avoid bias induced by stimulating only one pattern over and over again, each pattern in the network is stimulated at most once.

The number of stimulated HCs as well as the duration of the layer 4 stimulus is varied for different networks. For each stimulated HC 6 pyramidal cells in the MC belonging to the pattern are targeted.

¹each network has different seeds for the random number generators

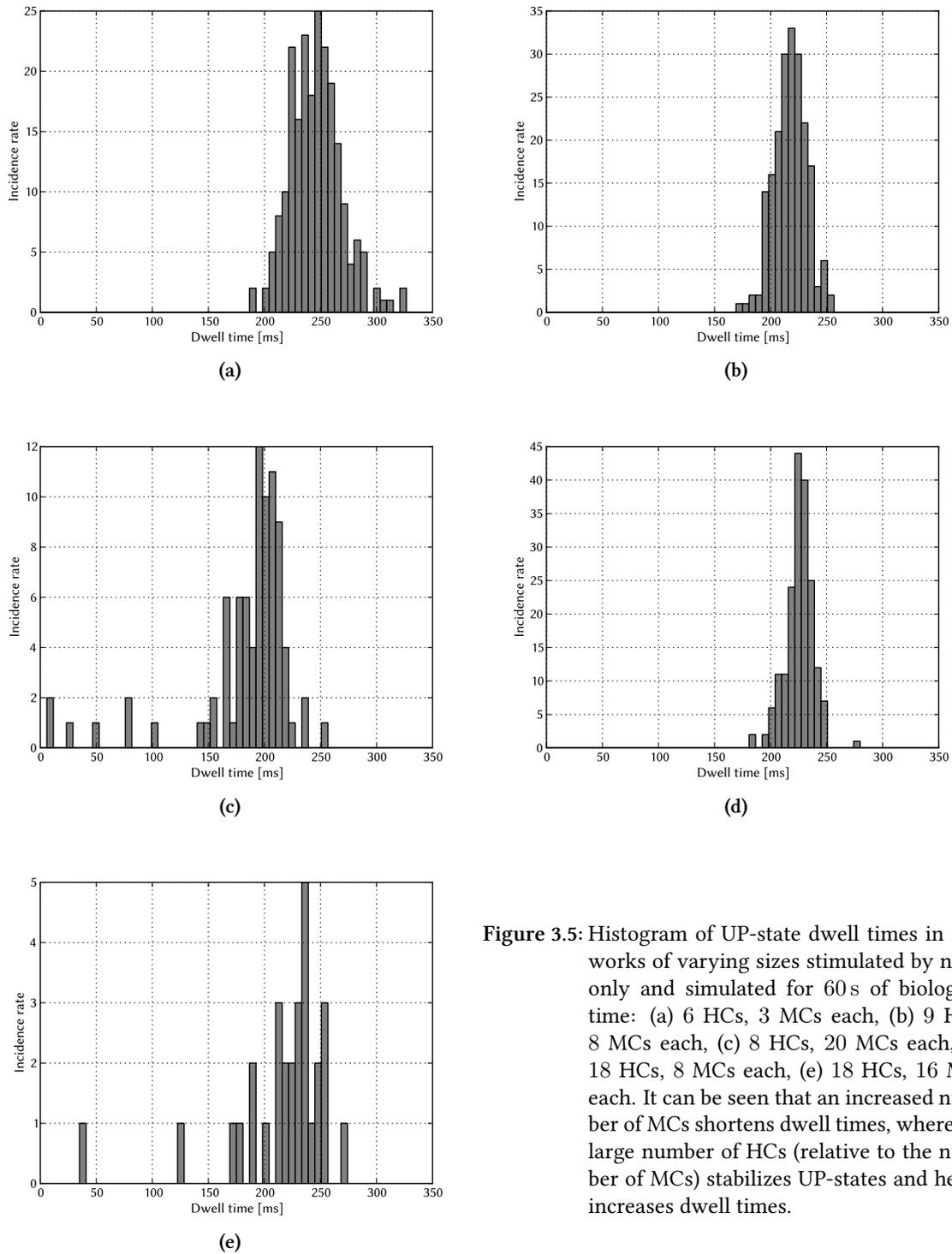


Figure 3.5: Histogram of UP-state dwell times in networks of varying sizes stimulated by noise only and simulated for 60 s of biological time: (a) 6 HCs, 3 MCs each, (b) 9 HCs, 8 MCs each, (c) 8 HCs, 20 MCs each, (d) 18 HCs, 8 MCs each, (e) 18 HCs, 16 MCs each. It can be seen that an increased number of MCs shortens dwell times, whereas a large number of HCs (relative to the number of MCs) stabilizes UP-states and hence increases dwell times.

3.2 Orthogonal Patterns

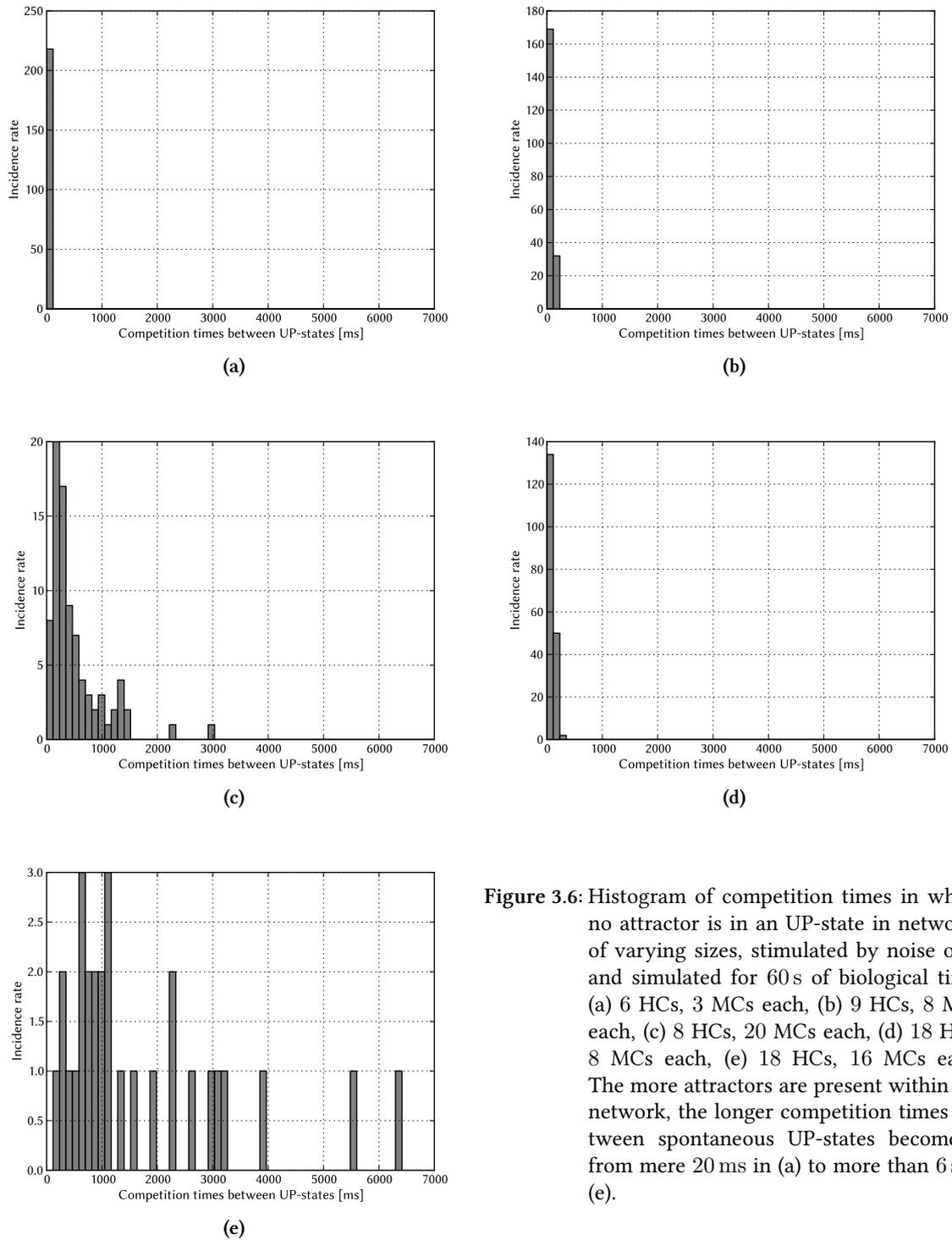


Figure 3.6: Histogram of competition times in which no attractor is in an UP-state in networks of varying sizes, stimulated by noise only and simulated for 60 s of biological time: (a) 6 HCs, 3 MCs each, (b) 9 HCs, 8 MCs each, (c) 8 HCs, 20 MCs each, (d) 18 HCs, 8 MCs each, (e) 18 HCs, 16 MCs each. The more attractors are present within the network, the longer competition times between spontaneous UP-states become – from mere 20 ms in (a) to more than 6 s in (e).

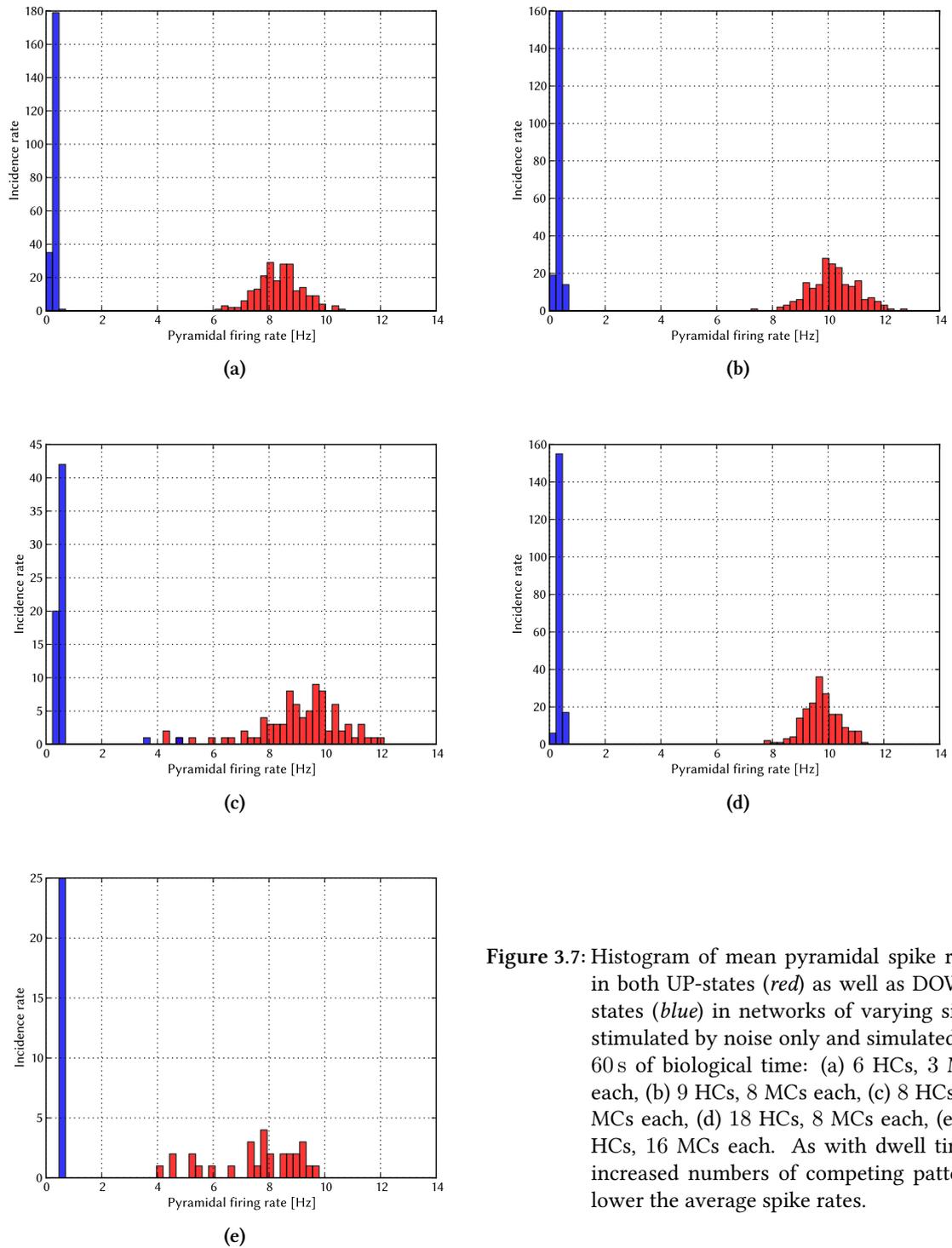


Figure 3.7: Histogram of mean pyramidal spike rates in both UP-states (*red*) as well as DOWN-states (*blue*) in networks of varying sizes, stimulated by noise only and simulated for 60 s of biological time: (a) 6 HCs, 3 MCs each, (b) 9 HCs, 8 MCs each, (c) 8 HCs, 20 MCs each, (d) 18 HCs, 8 MCs each, (e) 18 HCs, 16 MCs each. As with dwell times, increased numbers of competing patterns lower the average spike rates.

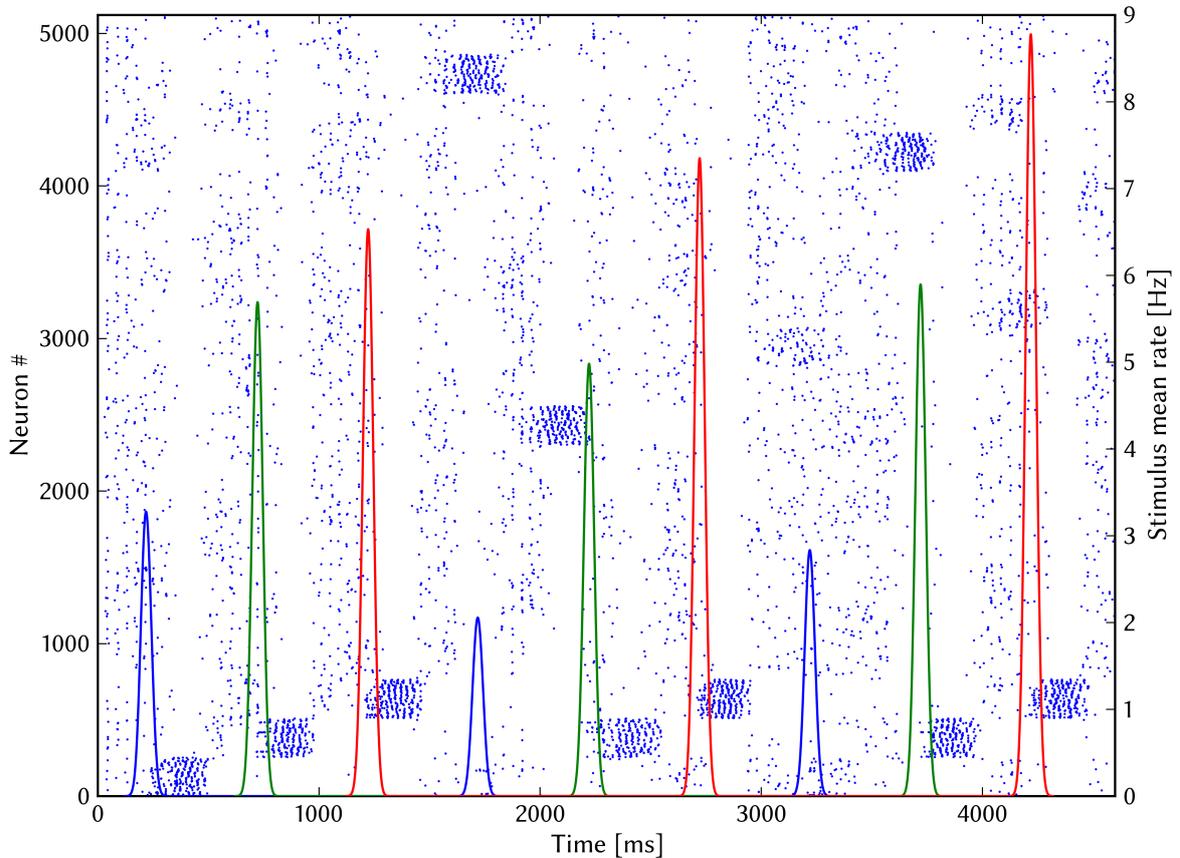


Figure 3.8: Pattern completion in a network with 8 HCs by 20 MCs. Pyramidal spikes and layer 4 input rate (colored by stimulated pattern) for the network in which three patterns 1, 2 and 3 are stimulated in 2, 4 and 6 out of possible 8 HCs in turn for 40 ms. The input rate is calculated over all layer 4 sources and normed to the total number of pyramidal cells in each pattern. This allows the comparison of total input strengths for different patterns because the stimulus rate is higher the more pyramidal cells of one pattern are stimulated in total. Since only 6 out of 30 pyramidal cells are stimulated in each MC that receives layer 4 input, the plotted stimulus rate is much lower than the 75 Hz set for the spike sources might suggest. Because the input is comprised of Poisson-distributed spikes (see 2.1.6 Stimulus), its total actual rate varies.

It is observed that stimulating either 2, 4 or 6 HCs for 40 ms is indeed enough to recall the whole pattern. Also, it can already be seen somewhat in this network that the probability of success is dependent on the number of stimulated HCs.

Note that this network is shown for demonstrating purposes only and was not included in the statistics gathered for figures 3.9 and 3.10.

Each configuration (HCs/ duration) is simulated in 4 different networks, amounting to 384 simulated networks in total.

The dynamics of such a network can be observed in figure 3.8. For easier visualization, only three patterns are stimulated repeatedly.

Afterwards, each network is analyzed. An activation attempt is said to be successful if the stimulated pattern is detected to be in an UP-state within 200 ms after the stimulus started. If another pattern

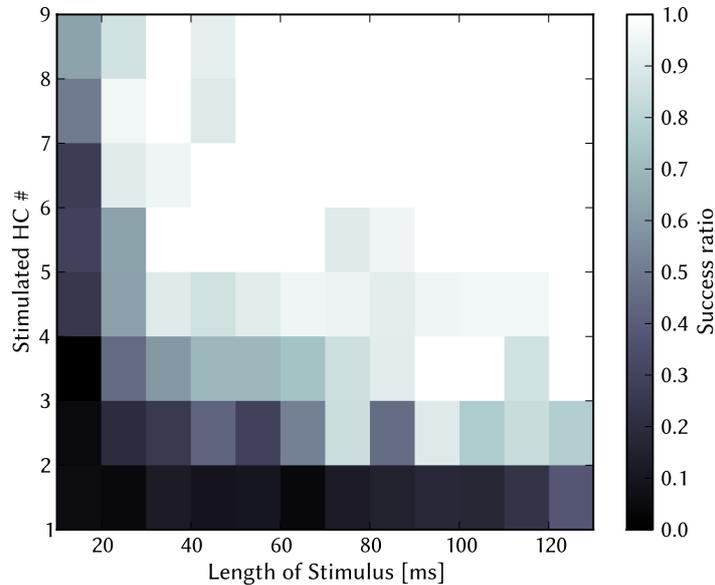


Figure 3.9: Pattern completion in a network of 8 HCs by 20 MCs. Success ratio for activating a specific pattern as described in the text. The color of each rectangle corresponds to the success ratio at the lower left corner. See also figure 3.10.

It is observed that once half of all HCs are stimulated for 30 ms or longer, pattern can be recalled quite reliably. On the other hand, stimulating only 1 out of 8 HCs or only for 10 ms is not enough to activate the attractor in most cases. Note that even in the case of 8 stimulated HCs pattern completion is performed, since only 6 out of 30 pyramidal cells per participating MC receive layer 4 stimulus.

is active during that time or if the stimulated pattern already was active up to 500 ms prior to activation the attempt is deemed invalid and ignored for calculating success ratios. This is done because it is far less likely to activate a pattern when another one is already active (see next section) or when the stimulated pattern is still suffering from adaption because it was active recently. Out of the 28 total attempts for each parameter combination, between 15 and 27 were valid. The rise time is defined as the time difference from the start of the stimulus until the first time the pattern is detected to be in an UP-state.

Both success ratios (figure 3.9) as well as average rise times (figure 3.10) are plotted. It can be safely concluded that the network is fully capable of pattern completion.

3.2.3 Pattern Rivalry

Another important feature is the inability of one pattern, stimulated by layer 4 input, to terminate another already active pattern and enter an UP-state itself. This phenomenon is the so-called *attentional blink*, studied by experimental psychologists (*Shapiro et al.* [1994], *Marois and Ivanoff* [2005]). When subjects are looking at different letters in rapid succession, if one letter is followed by another after a short enough interval, the second one is not perceived. *Lundqvist et al.* [2006] suggests

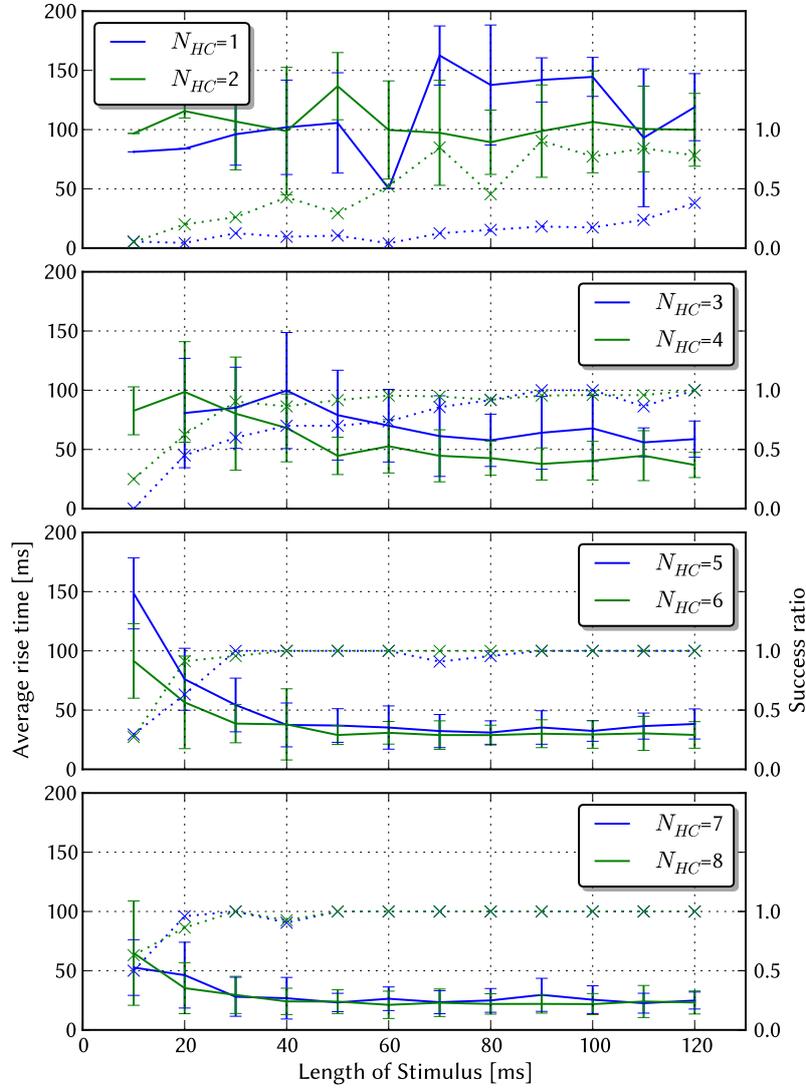


Figure 3.10: Average rise time for pattern completion in a network with 8 HCs by MCs (solid lines) for different numbers of stimulated HCs and stimulation lengths. The success ratio (dotted line) indicate from how many values averages and standard deviation (error bars) were computed and should not be confused with success ratios shown in figure 3.9. The data suggests that the average rise time is constant once patterns are reliable activated (half of HCs being stimulated for 30 ms or more) since there is a constant time that is needed for the activity to spread within each HC and throughout the pattern. Stimulating too few HCs leads to additional delays because the activity has to spread further (to more than half of all MCs); whereas too short stimulation is highly dependent on there being enough spikes in total and furthermore relies on advantageous background stimulation in order to achieve entering an UP-state.

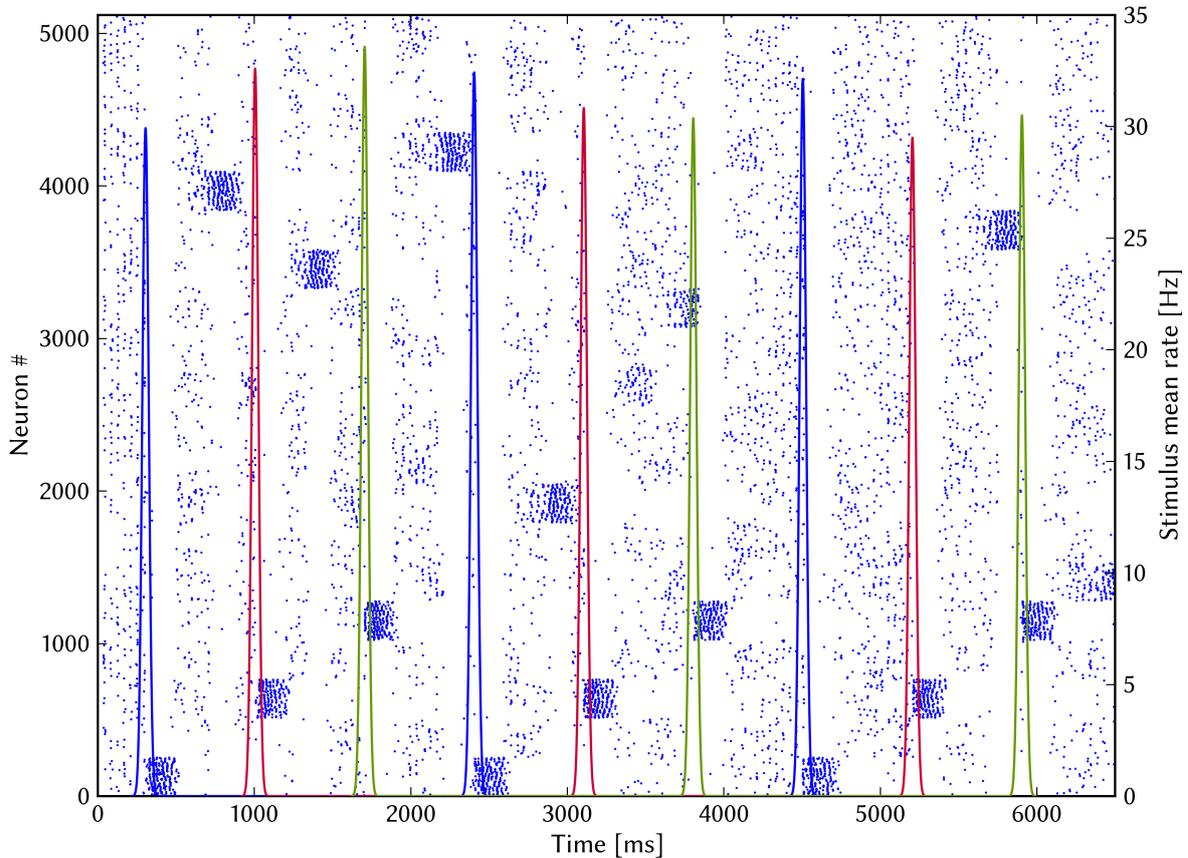


Figure 3.11: Demonstrating the reliable UP-state initiation of a network with 8 HCs by 20 MCs. Pyramidal spikes and layer 4 input rate (colored by stimulated pattern) are shown. Patterns 1, 3 and 5 are stimulated in the way described for pattern *A* in the text. The immediate onset of UP-states for this kind of stimulus can be seen. Already active spontaneous UP-states are successfully terminated. Note that because of filtering with a Gaussian (see 2.2.1 *Gauss Filter*) the very short input appears to be wider than it actually is. Also see figure 3.12.

Note that this network is only for demonstrating purposes and was not included in the statistics gathered for figure 3.13.

that this is due to the first letter triggering an attractor state which cannot be terminated by layer 4 stimulus corresponding to the second pattern (which in turn corresponds to the second letter).

This phenomenon is again investigated through a series of different networks of same size as in 3.2.2 *Pattern Completion* (8 HCs by 20 MCs). For each network, all 20 patterns are randomly assigned in pairs of two in shuffled order. Then, again in 700 ms intervals, 10 pattern rivalry attempts are made.

Let the two patterns in each pair be denoted *A* and *B*. In order to nearly guarantee an immediate UP-state of pattern *A*, *all* pyramidal cell receive stimulus with layer 4 like weights at 100 Hz for only 10 ms which amounts to one layer 4 spike per pyramidal cell on average. This provides a reliable way to have an UP-state start a defined point in time, which is all the more important, because UP-states are only about 200–250 ms long. Then, after a certain delay ΔT , pattern *B* is stimulated *regularly*, which means that 6 out of 30 pyramidal cells receive layer 4 input at 75 Hz for

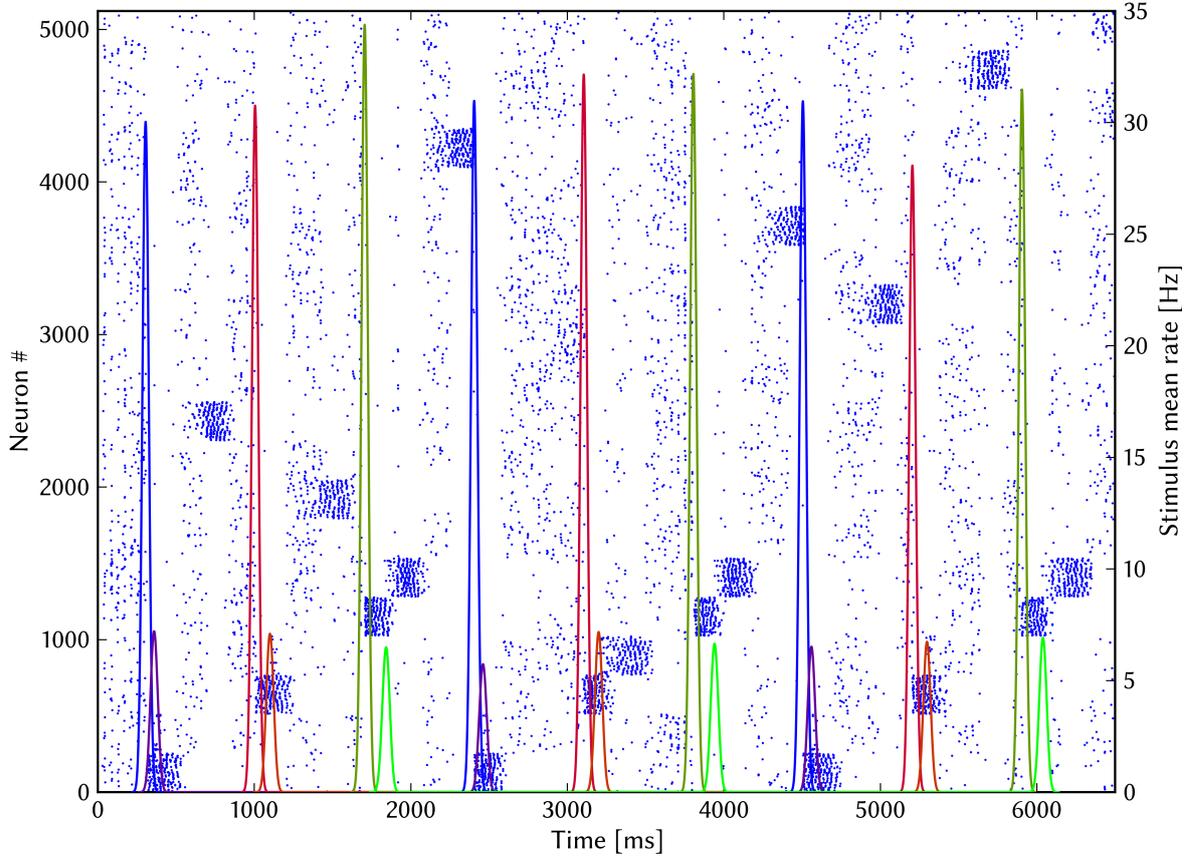


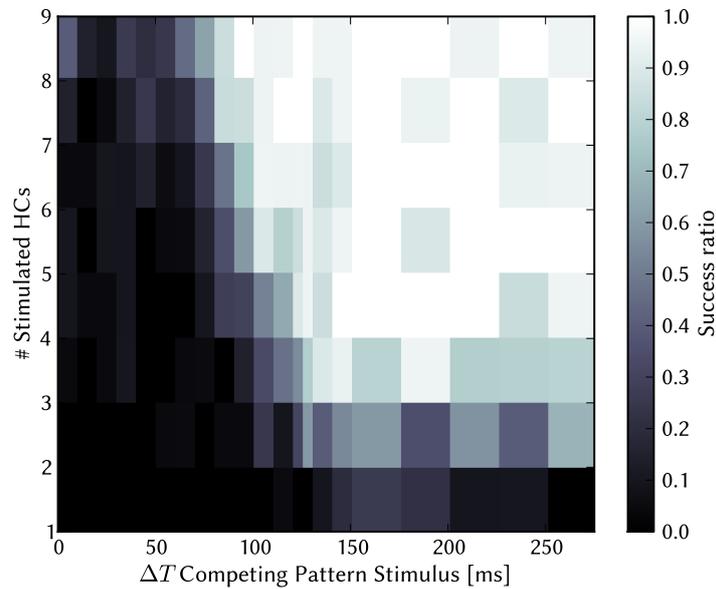
Figure 3.12: Network dynamics of the same network as in figure 3.11 where additionally patterns 2, 4 and 6 receive pattern B like stimuli (see text) to 5 HCs after 40 ms, 80 ms and 120 ms. The input rate is calculated over all layer 4 sources and normed to the total number of pyramidal cells in each pattern. This allows for the visualization of different input strengths.

It is clearly observed that even though the stimulus in itself is governed by the same parameters in each case, the time delay to the onset of the UP-state has a great influence as to whether the competing attractor succeeds. For 40 ms it is not possible, for 80 ms only one out of three attempts succeeds and for 120 ms all three attempts succeed. See figure 3.13 for more extensive statistics.

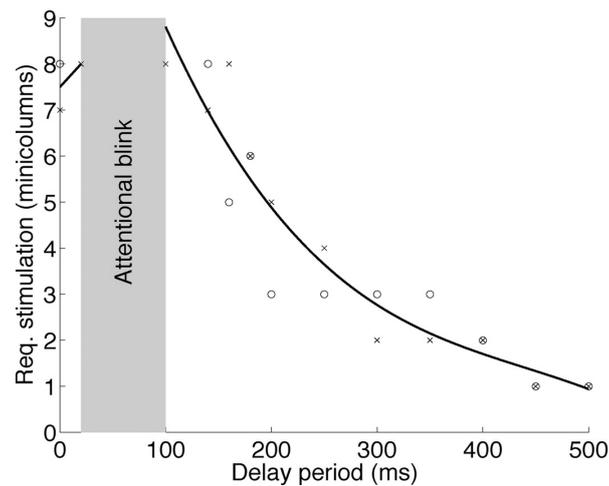
Note that this network is only for demonstrating purposes and was not included in statistics gathered for figures 3.13.

40 ms. The duration was chosen to be roughly $1/5$ of the average attractor dwell time, corresponding to the 120 ms stimulus while observing 500–1000 ms long UP-states (Lundqvist *et al.* [2006]). The number of stimulated HCs as well as the delay ΔT are varied for each network. Each combination of parameters is realized in two networks, totaling up to 336 separate simulations.

The same way as in 3.2.2 *Pattern Completion*, each network is then analyzed as to whether pattern B was successfully activated or not. If the competing pattern B was activated within 200 ms after the stimulus started, the attempt is counted as successful – otherwise it is deemed unsuccessful. Furthermore, the rise time from the beginning of the stimulus until the onset of the UP-state is recorded for each successful attempt. As before, attempts during which spontaneously activated patterns in-



(a)



(b)

Figure 3.13: (a) Pattern rivalry in a network of 8 HCs by 20 MCs. Success ratio for activating a specific pattern B when pattern A is already active as described in the text. The color of each rectangle corresponds to the success ratio at the (x, y) values of its lower left corner. See also figure 3.14. (b) Pattern rivalry as observed in the original model (figure 7 from *Lundqvist et al.* [2006]).

It is observed that roughly within the first 70–80 ms after pattern A was activated, stimulus for pattern B is highly unlikely to terminate pattern A 's UP-state. For larger ΔT success is more likely the more HCs are stimulated, as would be expected. Finally, when the stimulus of 40 ms is still active when an UP-state decays on its own (which is the case when its onset is roughly at 175 ms), the probability of activating pattern B returns to the value observed in figure 3.9, because same as when no pattern is active, the stimulated pattern B gets a head start once pattern A decays regularly.

Compared to the original model the observed time span of attentional blink is somewhat shorter with a steeper falling edge. This can be explained with the dwell times of UP-states which are shorter than in the original model in general.

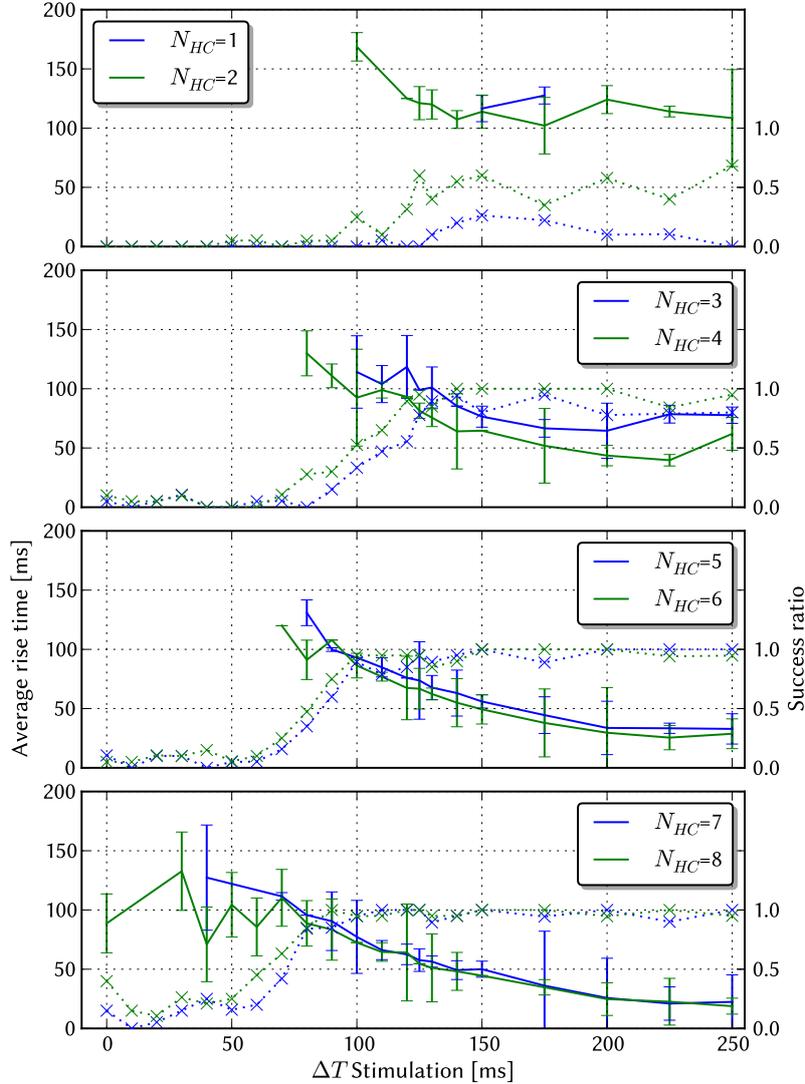


Figure 3.14: Average rise time for pattern rivalry in a network with 8 HCs by MCs (*solid lines*) for different numbers of stimulated HCs and delays ΔT between active pattern A and competing pattern B (see text). The success ratio (*dotted line*) indicate from how many values averages and standard deviation (*error bars*) were computed and should not be confused with success ratios shown in figure 3.13. If the success ratio was below 20 %, the corresponding rise times have been omitted.

It is observed that for low ΔT rise times are high and vary a lot, if they exist at all. This is due to the fact that the layer 4 stimulus alone is generally not enough to terminate the pattern. Therefore the UP-state is only terminated if by chance pattern A is weak and the stimulus to pattern B is concentrated enough. Furthermore, high rise times indicate that pattern B only became active after pattern A decayed regularly, since it had an advantage over all other patterns due to the received additional stimulus. The latter point also explains why rise times approach their regular pattern completion counterparts (observed in figure 3.10) for large ΔT .

tervene are ignored and therefore not counted as unsuccessful activation attempts. Overall between 16 and 20 valid attempts were recorded for each parameter combination. From all successful and unsuccessful attempts, the success ratios are calculated and plotted in figure 3.13. Rise times are plotted in figure 3.14. Both figures show that *attentional blink* is clearly present in the investigated network.

3.3 Hardware Imperfections

When porting the model to neuromorphic hardware, one has to deal with different constraints. With regular CPU-based architecture one is limited by the processing power, which means that the time needed to complete a network simulation scales with the size of the network. This is different in neuromorphic hardware – since it is inherently parallel, computational time does *not* scale with network size. However there are several other limitations to consider. These include the number of possible hardware neurons and synapses as well as the parameter ranges being physically limited. Furthermore, long range connectivity is limited by bandwidth accompanied by a possible loss of spikes. Also – as is the case with all analog circuitry – all components suffer from process variations which can be countered with calibration only to a certain extent.

In order to give an estimate as to how the network model presented here will react under various kinds of distortions, the three most important ones are investigated here via manipulation of parameters along with methods of compensation.

The network used for investigations was of default network size, 9 HCs by 8 MCs, in order to study the change in spontaneous network behaviour (no layer 4 input was applied). Each distorted network shown in the following sections, was simulated for 120 s of biological time. Since, as before mentioned in 2.3 *Implementation*, recording voltage traces is *very* memory consuming (compared to spike data), each network was again simulated for 5 s while recording voltages. Overall 99 configurations of the network are simulated in these two ways.

3.3.1 Neuron Loss

Since the number of hardware neurons is known in advance, it is an obvious for networks to be emulated to not exceed this number. However, if one needs to have a certain number of HCs and MCs which would to a violation of this restriction or the attractor-memory is part of a larger network to be emulated, it might be necessary to decrease the number of neurons residing in each MC. Since in the default setup pyramidal cells make up almost 91 % of all cells in the network, it is easiest to homogeneously reduce their numbers.

First, the decrease is not compensated (plotted in *red* in figure 3.16), whereas in another simulation run the connection probabilities as well as the weights were increased according to table A.4 in which N_{pyr} represents the reduced count of pyramidal cells per MC (plotted in *blue*). It can be seen that up to a loss of 20 % compensation is not really necessary as all investigated features remain quite similar (note that the difference in membrane potential could still be explained due to the short simulated biological time as well as the fact that only a small subsample of all pyramidal cells was recorded). Beyond that, regular network dynamics in the uncompensated case break down, evidenced by the increase in competition time, a decrease in dwell time as well as an overall

increase in DOWN-state spike rates (the increase in detected UP-states for 80 % and 90 % is merely an artifact of the UP-state detection method because overall spike rates are too low).

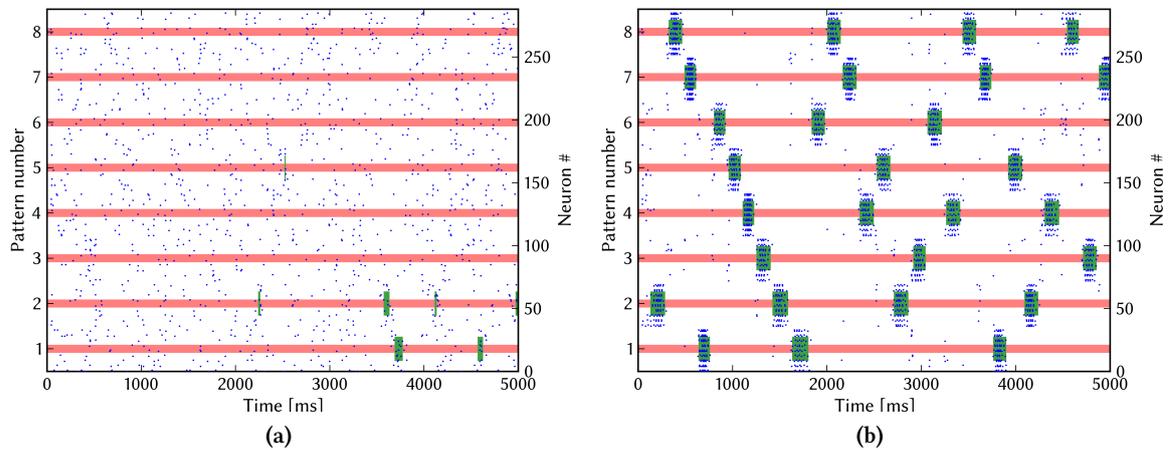


Figure 3.15: Network dynamics in a network of default size with 90 % neuron loss simulated for 5 s of biological time: Both the uncompensated (a) as well as the compensated case (b) are shown. Amazingly, even with a mere 3 pyramidal cells per MC, regular network dynamics are observed, whereas in comparison it is obvious that the slight concentrations of spikes in the uncompensated case can hardly be regarded as UP-states. The horizontal lines denote UP- (*green*) and DOWN-states (*red*) of the attractors and have been automatically generated by the analysis module.

By compensating for the lower neuron number, it is possible to preserve network dynamics for even as much as 90 % loss – see figure 3.15 for a comparison of network dynamics in that case. Because connection probability and synaptic weights are increased, both excitation during UP-states and inhibition during DOWN-states of a single cell increase, as evidenced by lower average membrane potentials during DOWN-states as well as higher average membrane potentials and spike rates during UP-states. The latter therefore lead to somewhat shorter UP-states, because the neuron parameters – including the ones governing adaption – were left unchanged.

It can thus be concluded that the network is very resilient to neuron loss when properly compensated for. Since a reduction of neurons is always accompanied by a reduction of synapses, it is a viable strategy when compensating for the effects of synapse loss, discussed in the next section.

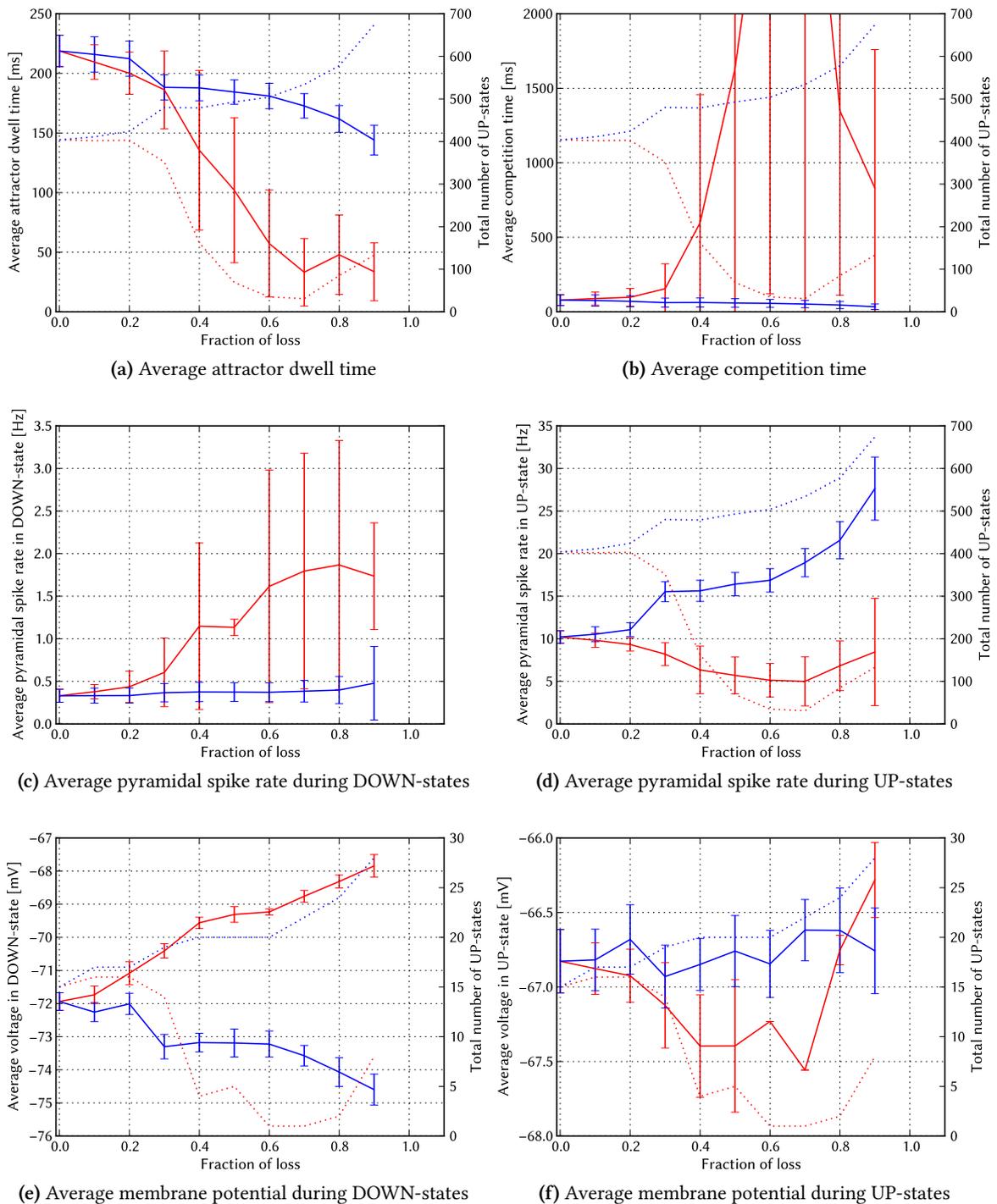


Figure 3.16: Network of default size for varying degrees of pyramidal neuron loss. Neurons were lost homogeneously across all MCs. Both the uncompensated case (*red*, all other parameters are left untouched) as well as rescaling according to the rules in table A.4 (*blue*) are shown. In order to give a sense of how many data points were used for computing the plotted values, the total number of UP-states (*dotted line*) is plotted as well. Stimulation time was 120 s for (a) to (d) and 5 s for (e) and (f). Refer to the text for further information.

3.3.2 Synapse Loss

Different than for neuron loss, it is not completely known prior to the process of mapping a given network onto the hardware how many synapses a given model may have. Furthermore, if synapses cannot be placed, they will most likely not be omitted homogeneously throughout the whole network. Still, because all processes concerning synapse loss are too complex to effectively simulate during this thesis, a severe simplification was made in that all synapse loss was taken to be homogeneously across all connection types. Therefore synapse loss was simulated by

$$\tilde{p} = p \cdot (1 - f) \quad (3.1)$$

where p is the original connection probability, \tilde{p} the connection probability after synapse loss and f the fraction of loss.

Again, as with neuron loss, the results of compensation (*blue*) as compared to the uncompensated case (*red*) were investigated in figure 3.17. Compensation was achieved by increasing the weights of all connections as to keep the synaptic input of each cell the same.

$$\tilde{w} = \frac{w}{1 - f} \quad (3.2)$$

where w is the original and \tilde{w} is the increased synaptic weight.

As is observed in figure 3.17, synapse loss – when uncompensated – is even more harmful to network dynamics than neuron loss. Already at 40–50 %, no dominant UP-states are observed though there are faint remnants (see figure 3.18a). At 90 % synapse loss no structure whatsoever can be identified (see figure 3.19a).

When compensated for, synapse loss of up to 40 % can be effectively countered with only minor distortions in the network behavior, such as the total number of UP-states which goes along with longer competition times (see figures 3.17 as well as 3.18b). Beyond that, the network structure remains recognizable but its characteristics become severely altered, such as dwindling dwell times. This can be explained by the fact that even though connection weights are increased, there are simply too few synapses in total for activity to spread and patterns to interact with each other. This leads to a situation where several patterns are in a semi-UP-state but none has the strength to completely suppress all others. Regular UP-states are only observed for very short times. Nevertheless, it is quite impressive that – when compensated for – the network activity pattern remains recognizable even with only 10 % of the normal synapses (see figure 3.19).

Therefore it can be concluded that, while not as resilient as to neuron loss, the network is capable of withstanding small to medium fractions of synapse loss.

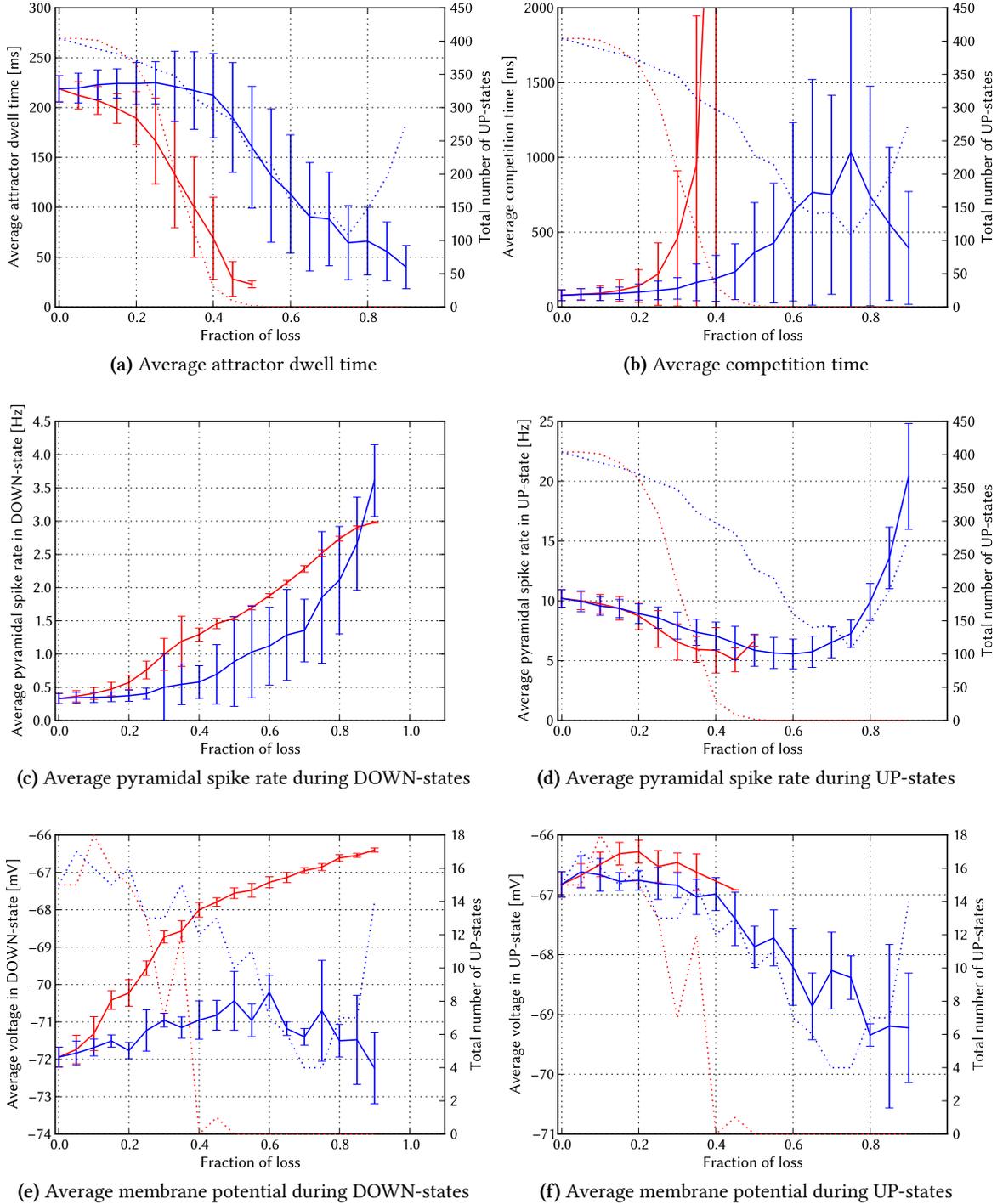
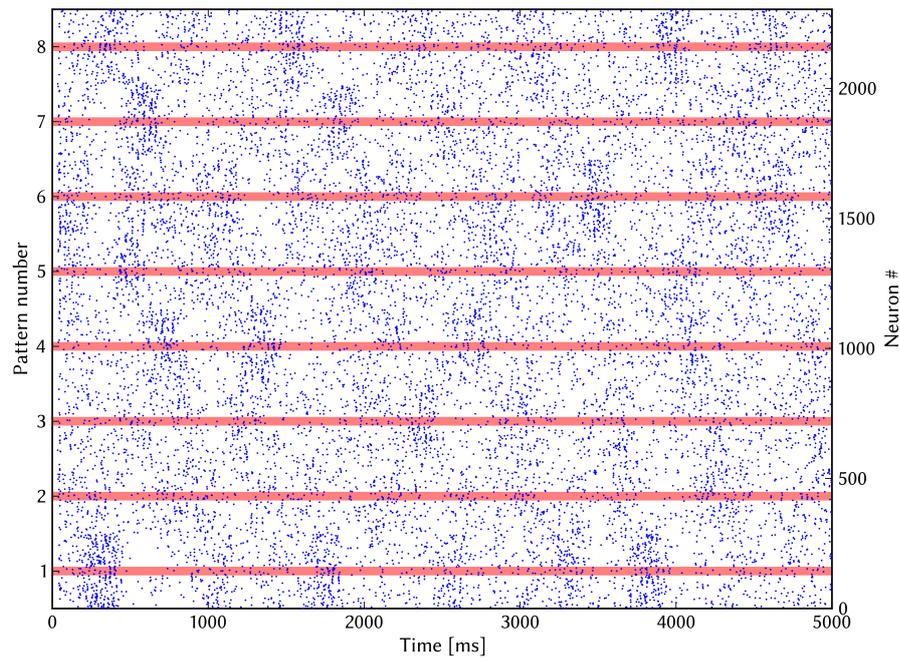
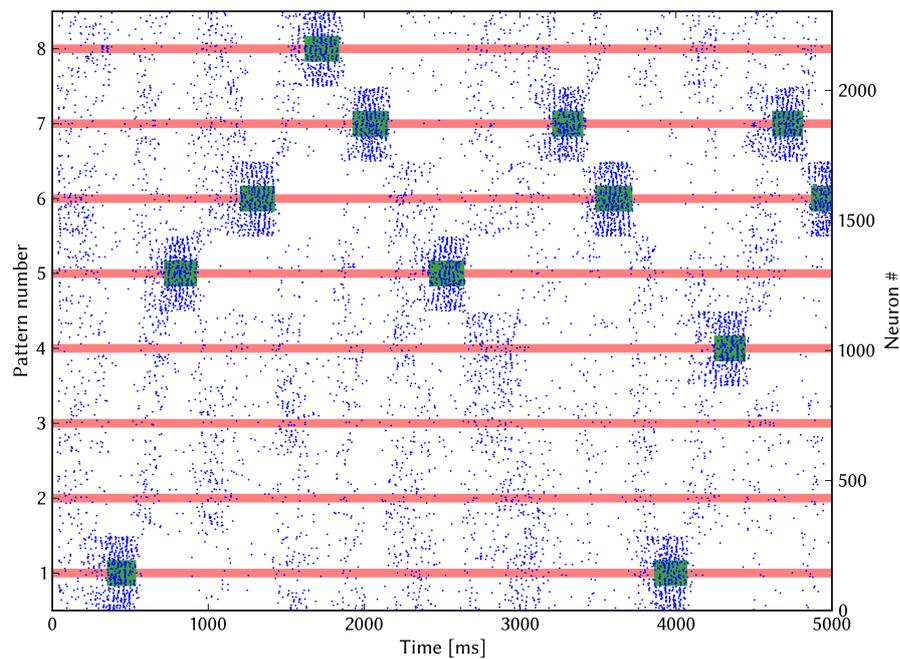


Figure 3.17: Network of default size for varying degrees of total synapse loss. Synapses were lost homogeneously among all connections by reducing connection probabilities according to $\tilde{p} = (1 - f) \cdot p$ where f is the fraction of loss. Both the uncompensated case (*red*) as well as increasing the weights according to $\tilde{w} = w \cdot (1 - f)^{-1}$ (*blue*) are shown. The total number of UP-states (*dotted line*) is plotted as well. Stimulation time was 120 s for (a) to (d) and 5 s for (e) and (f). Refer to the text for further information.

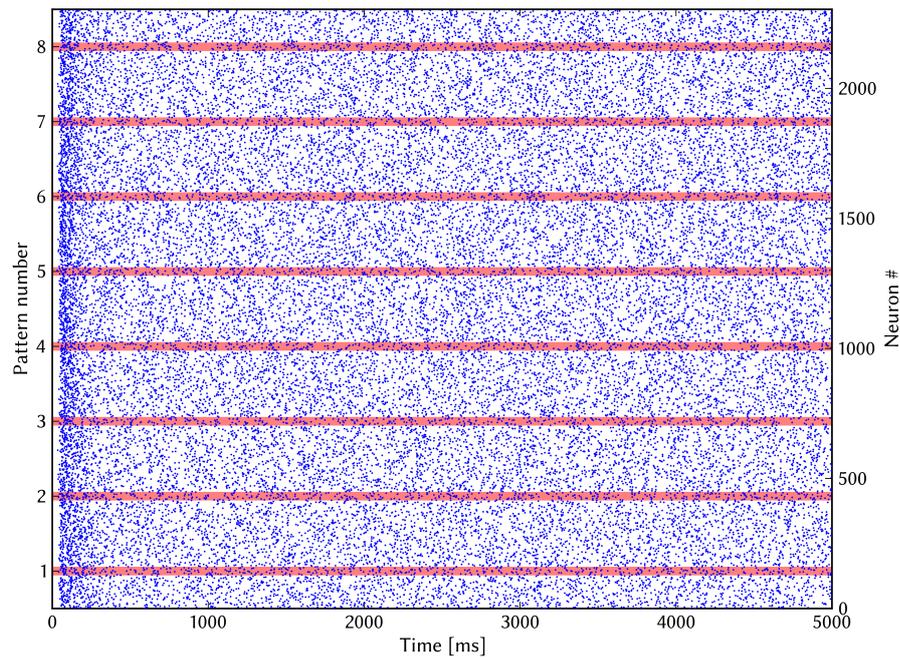


(a)

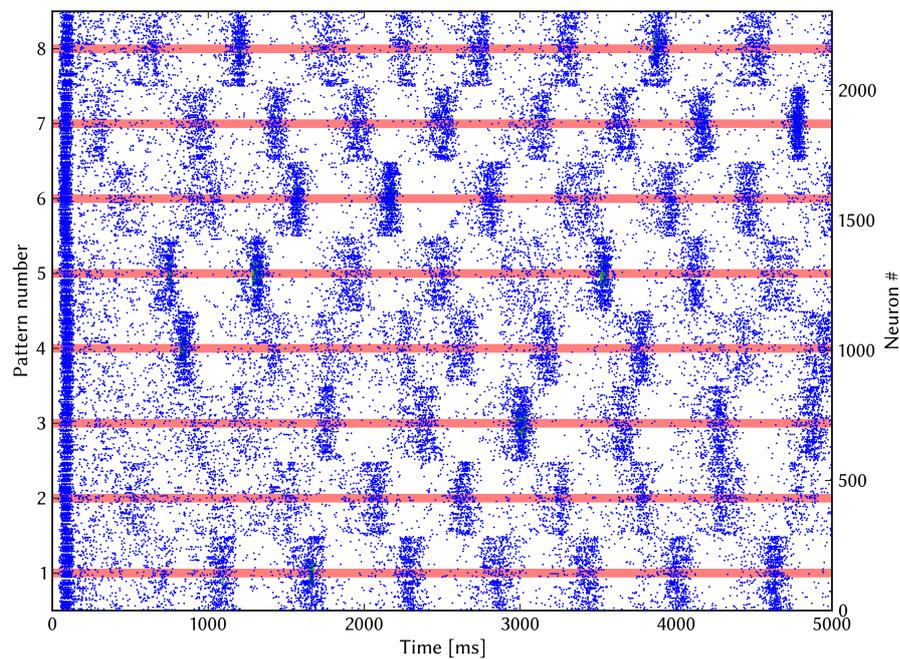


(b)

Figure 3.18: Network dynamics in a network of default size with 40% synapse loss simulated for 5 s of biological time: Both the uncompensated (a) as well as the compensated case (b) are shown. In the uncompensated case network dynamics are suppressed to the point that they fail to satisfy the set criteria for UP-states, but are still visible as slightly elevated spike rates. Furthermore, it can be seen that by means of compensation the regular network dynamics can be restored. The horizontal lines denote UP- (*green*) and DOWN-states (*red*) of the attractors and have been automatically generated by the analysis module.



(a)



(b)

Figure 3.19: Network dynamics in a network of default size with 90 % synapse loss simulated for 5 s of biological time: Both the uncompensated (a) as well as the compensated case (b) are shown. While the uncompensated case bears no resemblance to the original model whatsoever, network dynamics can be somewhat restored via compensation yet the observed periods of elevated spike rates within each pattern do not satisfy the set criteria for UP-states. See the text for further discussion. The horizontal lines denote UP- (*green*) and DOWN-states (*red*) of the attractors and have been automatically generated by the analysis module.

3.3.3 Fixed Pattern Noise

Fixed pattern noise in the distribution of synaptic weights is different from the previous two types of distortion in that it is not compensable as easily, because even though designed with the same specifications, analog circuitry is always subject to the tiniest variations which – even when calibrated – cause synaptic weights to be subject to variation as well. For an estimation of synaptic weight noise on an uncalibrated “Spikey” system, see Müller [2011]. While possible compensation mechanisms – such as increasing the average weights – do exist, they were not part of the investigations conducted in this thesis. It is therefore analyzed, how susceptible the network model is to varying connection weights. When simulating the effect of fixed pattern noise the actual weight of each single synapse present in the network was normally distributed around its regular value μ with certain width σ proportional to μ . Several different coefficients of variation $c_v = \sigma/\mu$ are explored. For improved statistics each c_v is simulated in four differently seeded networks.

From the data plotted in figure 3.20, it is seen that even noise up to $c_v = 20\%$ does not impede network operation significantly (see also figure 3.21a). Only the total number of UP-states is somewhat reduced and hence competition times increased which also causes average DOWN-state spike rates to rise – as one would otherwise see in a larger network (see figure 3.4 for comparison). Larger coefficients of variation bring the network dynamics to a near hold. Already at $c_v = 40\%$ only very few, rather short and low spiking UP-states occur.

It can therefore be concluded that small amounts of noise are tolerable. Further simulations in smaller networks show a size dependence of the influence on the network dynamics by hardware imperfections. As it turns out, smaller networks are more resilient than the network of default size investigated here. See the appendix B.1 for further data.

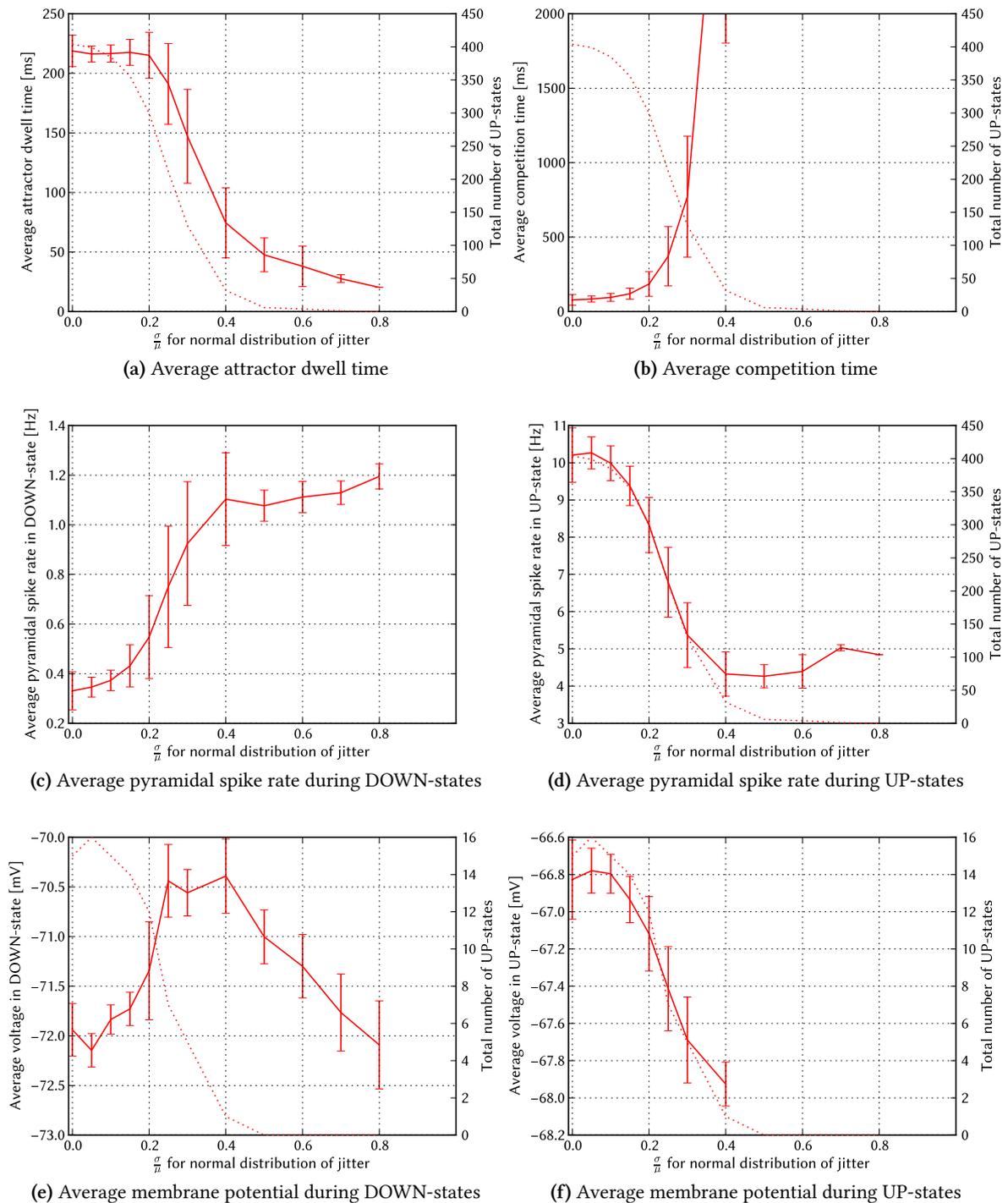
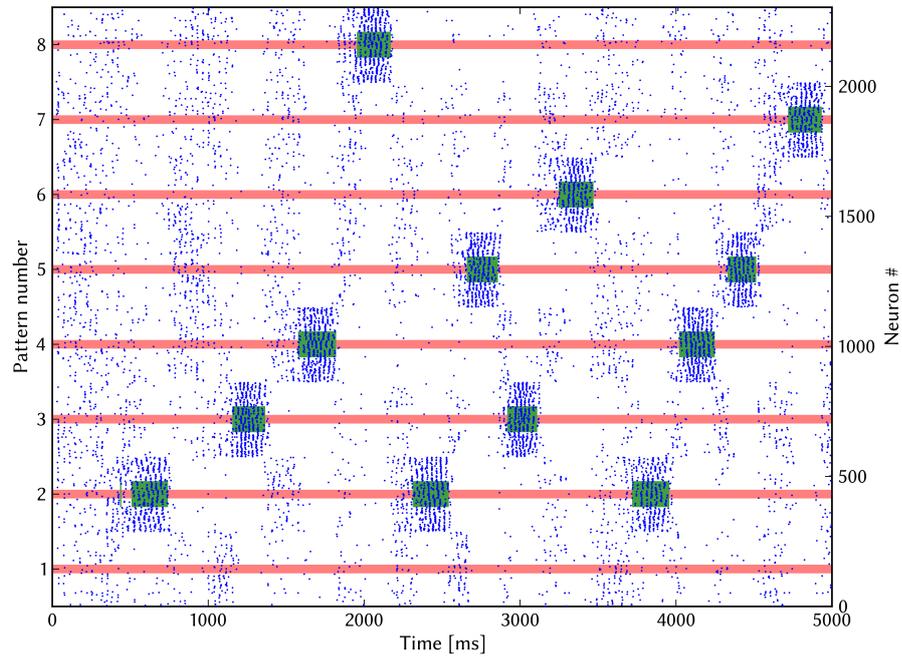
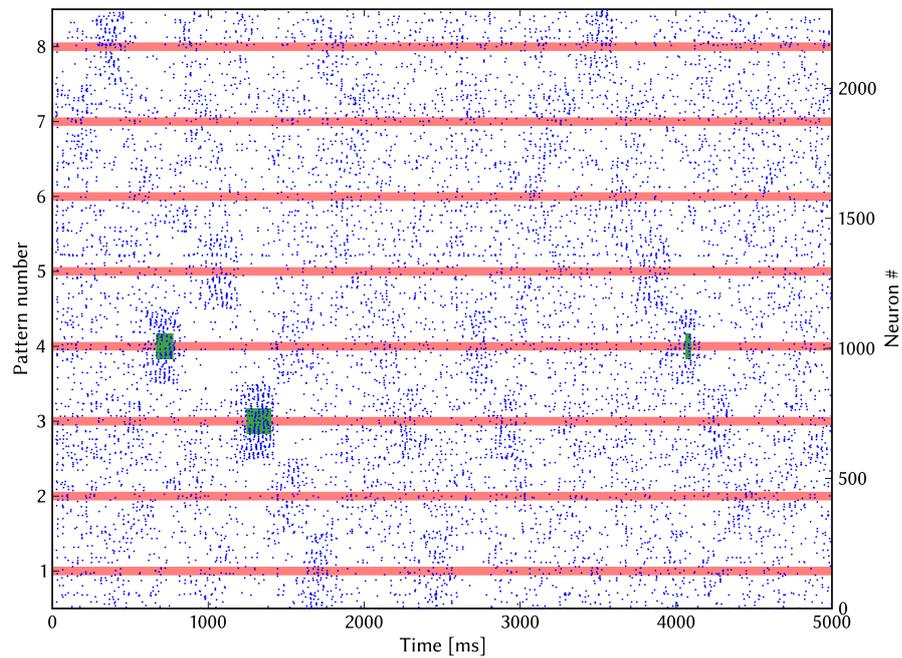


Figure 3.20: Network of default size for varying degrees of synaptic weight fixed pattern noise. Synaptic weights for *all* connections are distributed normally around their original values with varying widths σ . For each fraction of jitter σ/μ four different networks were simulated. In order to give a sense how many data points were used for computing the plotted values, the total number of UP-states (*dotted line*) is plotted as well. Stimulation time was 120 s for (a) to (d) and 5 s for (e) and (f). Refer to the text for further information.



(a)



(b)

Figure 3.21: Network dynamics in a network of default size with (a) $\sigma/\mu = 20\%$ and (b) $\sigma/\mu = 40\%$ normally distributed synaptic weights simulated for 5 s of biological time: It is observed that the network is capable of tolerating 20 % noise with only minor distortions (see text), while 40 % almost fully suppress spontaneous activation. The horizontal lines denote UP- (green) and DOWN-states (red) of the attractors and have been automatically generated by the analysis module.

3.4 Non-Orthogonal Patterns

Up to this point, all networks consisted of orthogonal patterns. Every MC took part in a single pattern. Now this restriction is lifted and the consequences observed.

The simplest way of introducing additional patterns is to take one MC from each original pattern. These patterns can be aligned diagonally as shown in table 3.1, and are henceforth called *diagonal patterns*.

	MC:	1	2	3	4
HC: 1		1,5	2,6	3,7	4,8
2		1,8	2,5	3,6	4,7
3		1,7	2,8	3,5	4,6
4		1,6	2,7	3,8	4,5

Table 3.1: Illustration of diagonal patterns. In a network of 4 HCs by 4 MCs, the patterns in which each MC participates are noted. In addition to the regular orthogonal patterns in which the index of the MC within the HC coincides, diagonal patterns are introduced in which the participating MC's index is advanced by one per HC.

A network with one diagonal pattern introduced is investigated in figure 3.22. The first and most important observation is the fact that regular network dynamics are preserved: It is still a single pattern that is in an UP-state rather than two patterns that overlap at the same time. However, the dwell time for the newly introduced pattern is significantly lower.

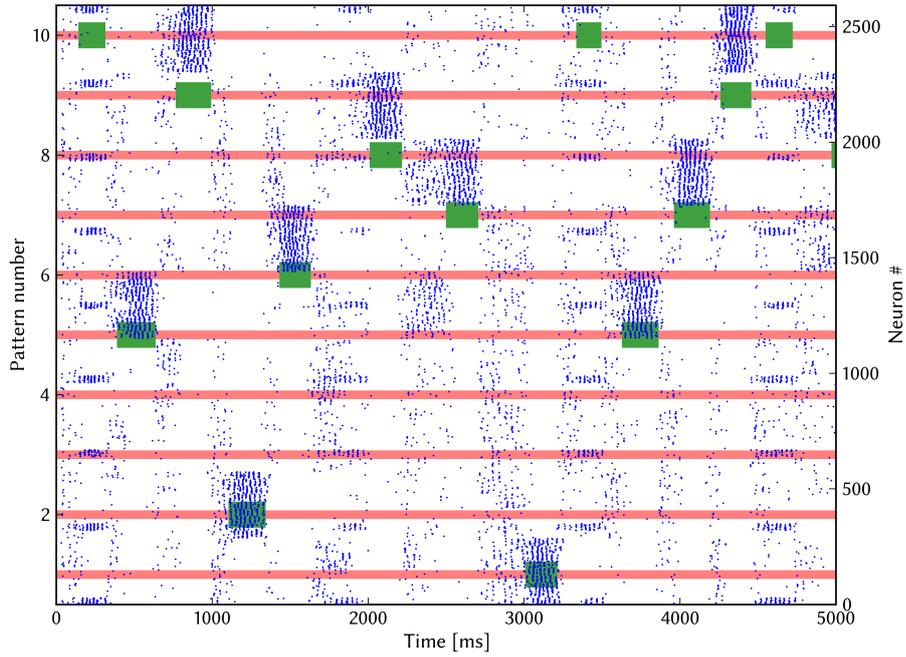
When patterns overlap, the network principles are changed in a significant way: In the orthogonal case, whenever an attractor is in an UP-state, all pyramidal cells in the other MCs are actively suppressed and receive no excitatory input except for noise and potential layer 4 stimulus. This changes in the non-orthogonal case. Since excitatory connections exist between any two MCs that share at least one pattern, MCs that are not part of the active attractor still receive excitatory input. However, as the majority of active MCs do *not* share a pattern with any given MC, the majority of input for any non-active MCs is still mostly inhibitory, hence the UP-state is able to emerge. If each MC is in several patterns, the ratio of inhibitory to excitatory connections between MCs for networks with the same number of HCs and MCs is constant, whereas, for other network sizes the ratio depends on which pattern is active and on the non-active MC in question.

If a network is made up of non-orthogonal patterns, MCs in the active pattern participate in other patterns as well. Since this increases the average spike rate, it lowers the standard deviation between all patterns' spike rates σ and hence distorts the UP-state detection mechanism presented in 2.2.3 *Spike-based*. A heuristic solution is to adjust the numerical constant c in equation (2.13) by the average pattern overlap (e.g. if all MCs participate in 2 patterns, $c = 2$). This is obviously only a solution for relatively small, evenly distributed overlap and increases the number of false-positive UP-state detections².

An interesting feature to look at are the transition probabilities governing the order in which attractors enter UP-states. For networks in which the number of HCs is an integer multiple of the number of MCs with 2 evenly distributed³ patterns, one observes that the network spends long periods of time in one of two subspaces (see figure 3.24). One set consists of the original, orthogonal patterns while the other one is made up of the newly introduced *diagonal* patterns (which in themselves are

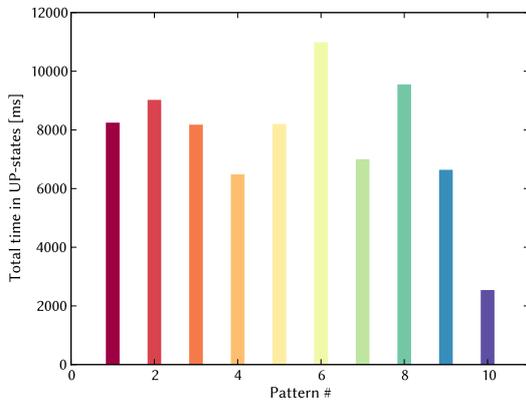
²As noted in 2.2.3 *Spike-based*, these can be filtered out since they are very short compared to regular UP-states.

³No two MCs share more than one pattern, see section B.2 for examples.



(a)

Figure 3.22: (a) Network dynamics of a network of 9 HCs by 9 MCs which had one diagonal pattern (index 10) introduced. The horizontal lines denote UP- (green) and DOWN-states (red) of the attractors and have been automatically generated by the analysis module. As one can see, the newly introduced pattern enters UP-states just like any other. Since it shares a MC with every other pattern, it is subject to far greater competition, limiting the dwell time of its UP-states in comparison to the other patterns. This is further evidenced by the total attractor dwell times for the same network simulated for 120 s shown in (b) where the newly introduced pattern (index 10) is active significantly less in total.



(b)

orthogonal as well). If the network is in an UP-state of either pattern, the next UP-state is far more likely to be of the same subset of patterns (see figure 3.24b).

This can be explained with the way patterns are set up here. Let the two subspaces be denoted by A and B . If a pattern of A is active, one MC of each pattern of B is active as well. Due to synaptic adaption, all these MCs need a refractory period until they can be activated once again. This means that all patterns of B have one MC less to compete for activation within the HCs while all other patterns from A are, in a sense, fully rested – hence they have an advantage in competition. This

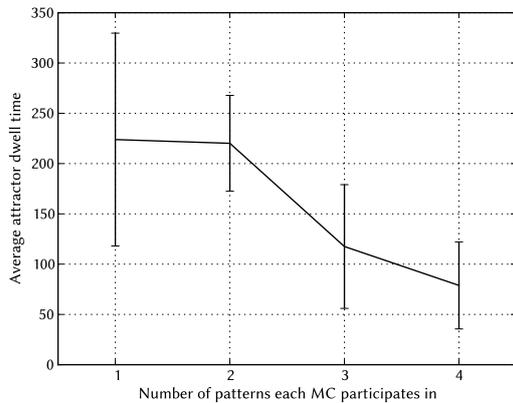


Figure 3.23: Average attractor dwell time for a network with 8 HCs with 8 MCs each simulated for 120 s of biological time. MCs participate in different numbers of patterns according to the tables in section B.2. It becomes apparent that the increased competition as well as stimulation of other patterns during an UP-state (see text) results in a shortening of attractor dwell times. All four networks were analyzed with a scaling factor of $c = 2.5$ for UP-state detection; competition times under 50 ms were ignored.

results in a higher transition probability to patterns within the same subset. Although not shown here, it was possible to control which meta-attractor the network was in by stimulating one of the corresponding patterns.

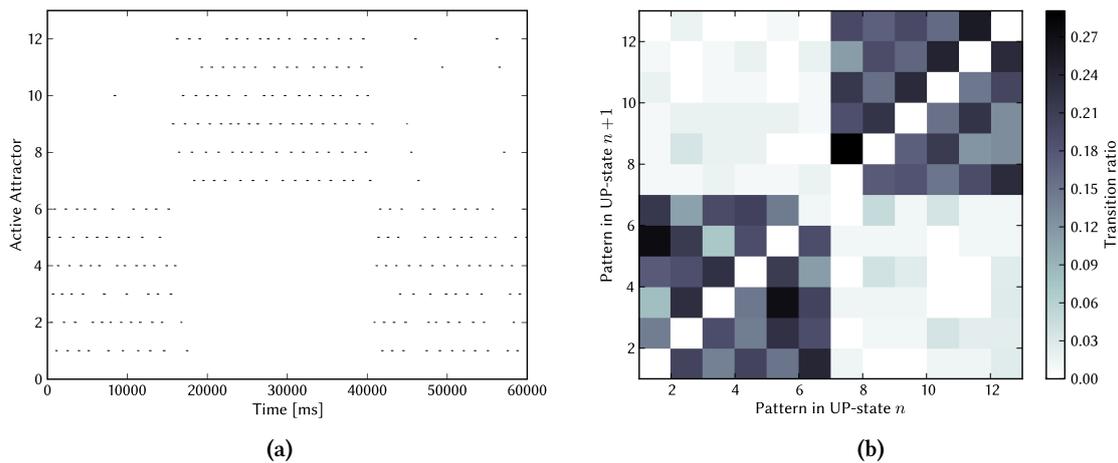


Figure 3.24: Network with 6 HCs by 6 MCs in which each MC participates in 2 patterns. (a) Active attractors denoted by horizontal lines over 60 s of biological time. One observes meta-attractors in which primarily patterns from one orthogonal subspace are activated. (b) Transition rates in the same network, simulated for 300 s of biological time. A transition was said to occur if UP-state n follows UP-state $n + 1$ within 500 ms. One clearly observes that transitions within the same orthogonal subspace are preferred.

3.5 Retinotopic Pattern Completion

Retinotopy describes the spatial organization of the neuronal responses to visual stimuli. Visual patterns observed correspond to the spatial pattern of activated neurons. So called retinotopic maps are found in several parts of the brain (Wandell *et al.* [2005]).

3.5 Retinotopic Pattern Completion

In order to implement retinotopic pattern completion, very simple images are encoded onto the different – now again orthogonal – patterns of the network. Each HC in this very simplified retinotopic map is said to represent a pixel in a picture. The MCs in each HC are then assigned the color (black/white) that the corresponding picture has in the respective pixel. This principle is illustrated in figure 3.25 with a sample encoding of digits. The network structure is changed from the original hexagonal to a rectangular grid with $500\ \mu\text{m}$ between nearest neighbours, representing the shape of the picture⁴.

Layer 4 stimulus is applied to the network in terms of pictures as well. Since each HC represents a pixel, it is assigned the color of the corresponding pixel in the input image. The stimulus then targets again 6 out of 30 pyramidal cells in all MCs whose previously assigned color coincides with the color of the HC they reside in. This leads to several patterns receiving layer 4 input *at the same time*. The aim then is to recall the pattern corresponding to the image observed.

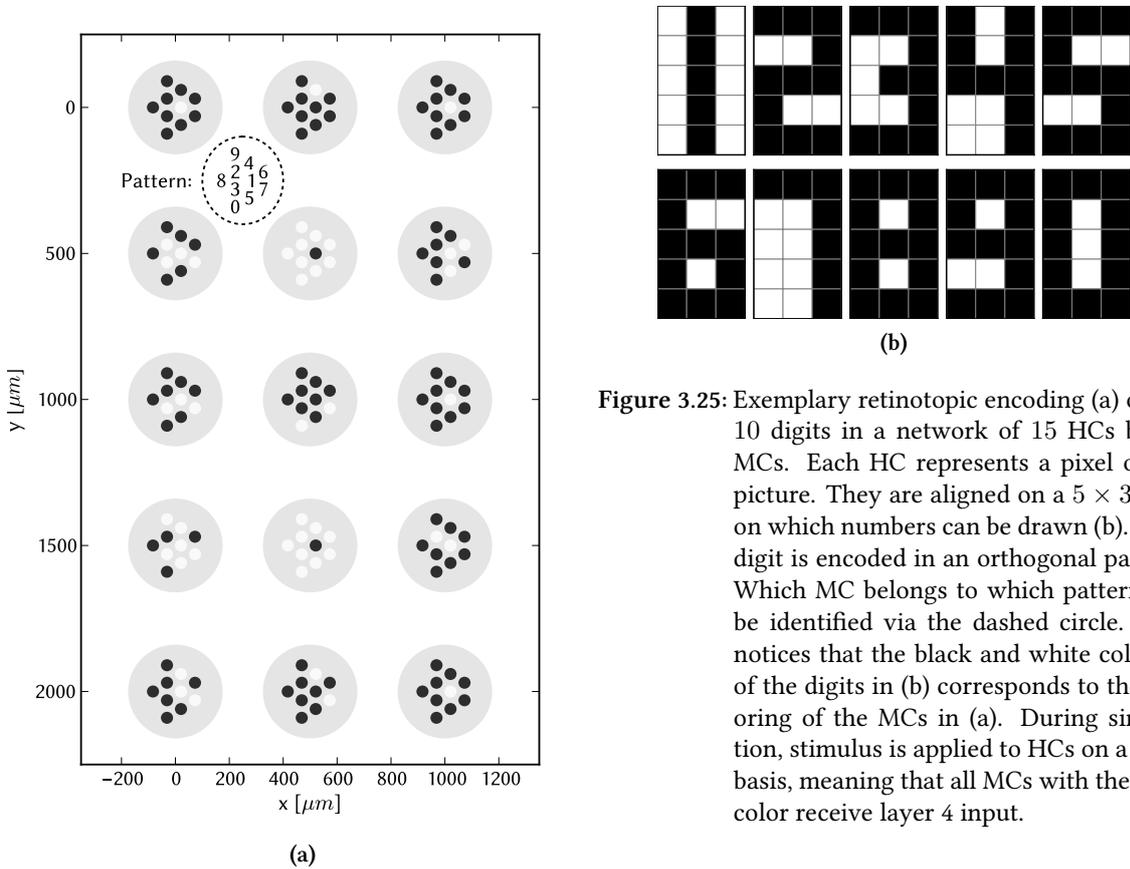


Figure 3.25: Exemplary retinotopic encoding (a) of the 10 digits in a network of 15 HCs by 10 MCs. Each HC represents a pixel of the picture. They are aligned on a 5×3 grid, on which numbers can be drawn (b). Each digit is encoded in an orthogonal pattern. Which MC belongs to which pattern can be identified via the dashed circle. One notices that the black and white coloring of the digits in (b) corresponds to the coloring of the MCs in (a). During simulation, stimulus is applied to HCs on a color basis, meaning that all MCs with the right color receive layer 4 input.

While the encoding of digits serves well as an intuitive example, it has several drawbacks when actually simulated in small networks such as those discussed in this thesis: Apart from digit 1, all other 9 digits share 6 pixels with each other⁵ and are overall quite similar (e.g. except for digit 4, all other digits share the middle black pixel in the topmost row). All this adds up to the fact that if

⁴This influences delays within the network, see 2.1.4 Delays

⁵They share all black corners except for the lower left, the middle black pixel in the rightmost column and the two white pixels from digit eight.

one digit is stimulated, several others will receive almost equal amounts of stimulus (e.g. digits 8 and 9, who only differ in a single pixel). One observes that for activating patterns in a retinotopic way, whether or not one pattern is activated instead of another is far more dependent on the ratio of stimulated MCs than the actual difference (e.g. the one MC difference between digits 8 and 9 is not enough when 14 and 15 MCs are stimulated in total). A first attempt at investigating this is made in figure 3.26.

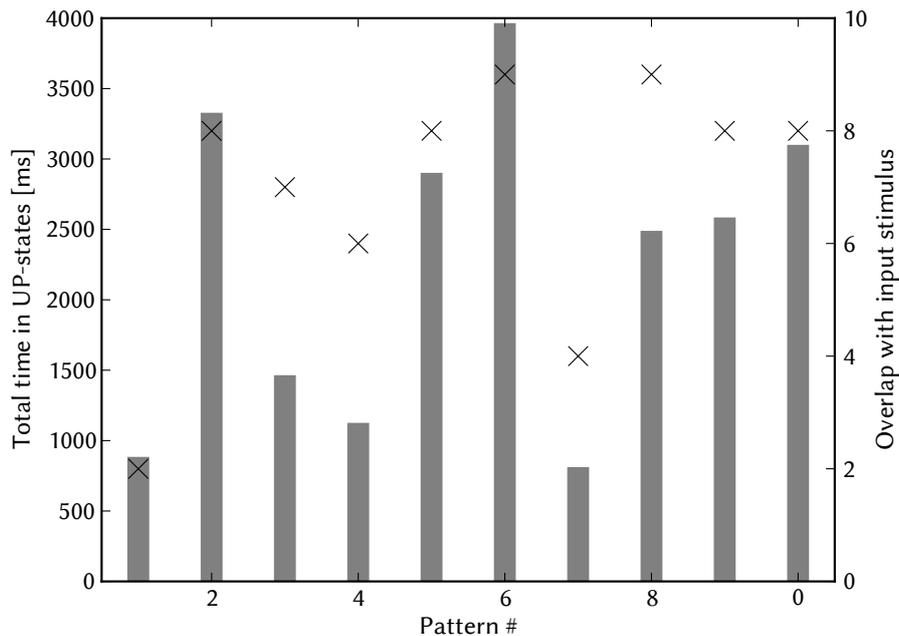


Figure 3.26: Total time spent in UP-states for each pattern in a network of 15 HCs by 10 MCs simulated for 60 s of biological time. The network was stimulated for 80 ms every 500 ms with an input image corresponding to the lower three rows of the digit 8 (the upper two rows receive no input at all). The number of MCs in each pattern that overlap with the input image (i.e. that receive stimulus) are denoted by crosses. It can be seen that while there is no exact one-to-one correlation between stimulated number of MCs and total dwell times, all patterns that have less than 7 MCs stimulated are significantly less active in total than those with 8 or more MCs stimulated.

Instead of digits, it is far more practical for small networks to encode more dissimilar. In order to simulate with a network of default size – and thereby shortening the time needed to complete a simulation – the switch to a 3×3 grid is made on which 8 images are encoded that differ in at least two pixels (see figure 3.27b). As seen in figure 3.27, the images are indeed dissimilar enough to recall the pattern corresponding to the input image.

While a more systematic investigation is currently being performed, it was shown here that the network is capable of retinotopic pattern completion.

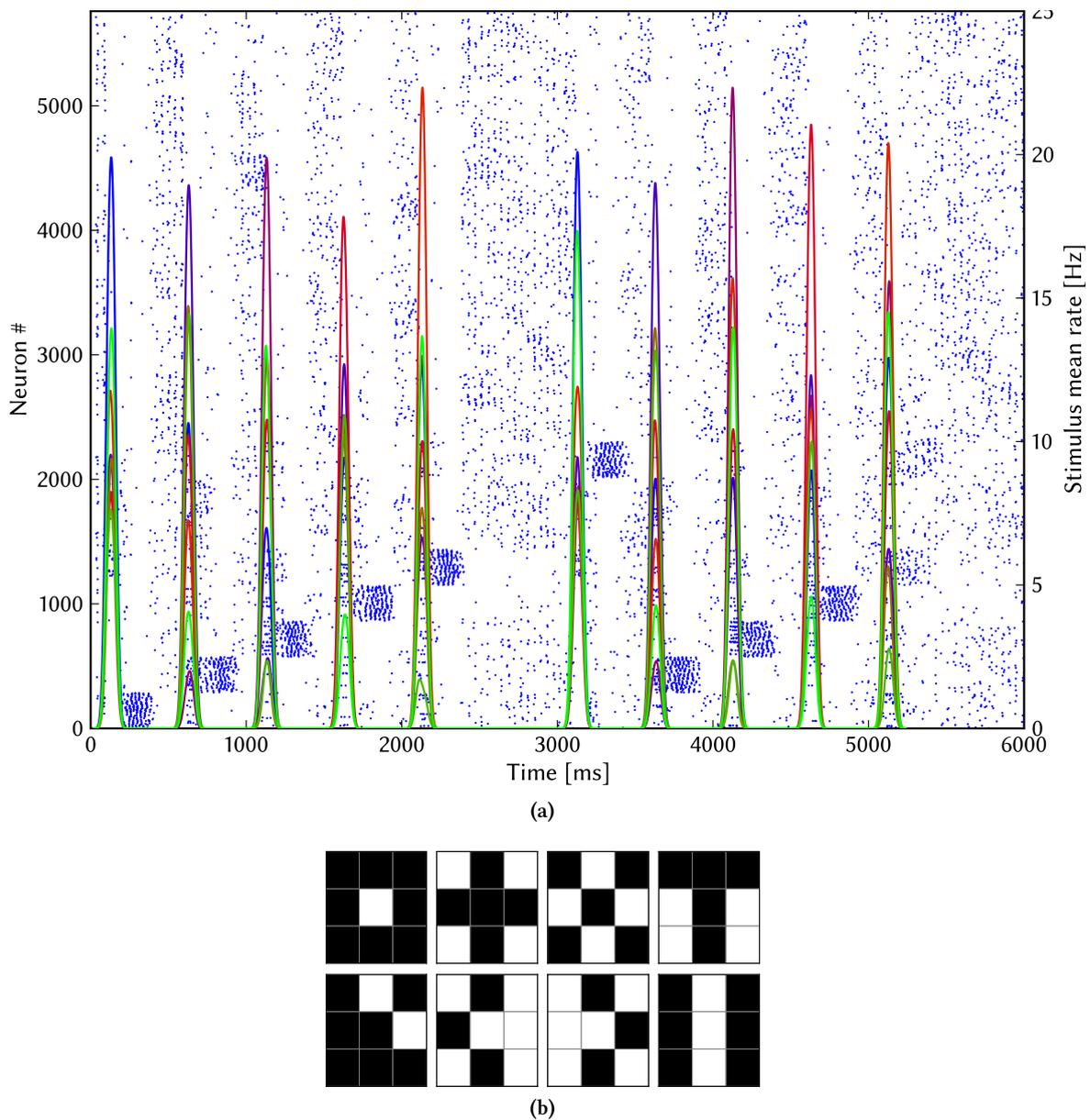


Figure 3.27: Retinotopic pattern completion on a separate set of images. (a) The dynamics of a network of size 9 HCs by 20 MCs where the first 8 of the 20 orthogonal patterns were encoded with the 3×3 images shown in (b). In contrast to the encoded digits (see figure 3.25) the images in this set are relatively more dissimilar to one another. Besides the encoded 8 patterns in (b), 12 other patterns that receive no layer 4 input at all were introduced in order to reduce the occurrence of spontaneous UP-states.

The patterns 1–5 (corresponding in (b) to the top row plus the first on the left in the second) were stimulated in ascending order for 60 ms in 500 ms intervals – the whole procedure was repeated twice. As one can see, stimulating an image results in several of the patterns receiving layer 4 input proportional to their similarity with the presented image to 6 out of 30 pyramidal cells. It is also observed, that most of the stimulated patterns are recalled. However, when the first image is stimulated for the second time, pattern 8 is activated, which is – as seen in (b) in the lower right corner – quite similar to the stimulated image 1.

4 Discussion and Outlook

The aim of this thesis was the investigation of a cortical attractor-memory network. It was shown that a neuromorphic hardware-compatible, modified version of the model proposed in *Lundqvist et al.* [2006] is indeed exhibiting the same phenomena as the original model. Both elevated spike rates in one attractor during an UP-state and almost no spikes in all other attractors are observed. For spontaneous activity, network features such as attractor dwell time and pyramidal spike rates in both UP- and DOWN-states are rather well preserved when the network size is changed and connection probabilities are scaled accordingly. When increasing the network size, competition times increase and spontaneous UP-states become less frequent, which is a desired effect when investigating the following features.

Both pattern completion as well as pattern rivalry in the network were examined. If no attractor is active, small amounts of layer 4 stimulus (20 % of all pyramidal cells stimulated for roughly $1/8$ of the average attractor dwell time in half of all MCs) are already enough to reliably induce an UP-state. When dealing with competing patterns, the so-called *attentional blink* phenomenon was successfully observed.

Furthermore, the influence of several distortion mechanisms relevant for the emulation on neuromorphic hardware were investigated. It was shown that the network can tolerate neuron loss very well with proper rescaling. Therefore, neuron loss could be used to counter the effects of synaptic loss which was shown to be compensable by weight adjustments only up to a certain point. The most severe threat to network dynamics is synaptic weight noise, at least for a network of default size and only for spontaneous UP-states. Further simulations revealed that its effect was far less pronounced in smaller networks.

By introducing non-orthogonal patterns, the memory capacity of the network was expanded. It was observed that the network is indeed capable of regular UP-state dynamics while each MC participates in more than one pattern. Of particular interest were the transition probabilities between UP-states, which should be investigated further. They provoke the emergence of *meta-attractors* during which the network – in which each MC participates in two patterns – is primarily in an UP-state of one of two orthogonal subspaces. So far, the concept has not been extended to more than two orthogonal subgroups. Also, connections between MCs in different HCs so far remained binary in nature (either purely excitatory or purely inhibitory) – if the number of patterns per MC was increased significantly, the strength of their connections could be made dependable on the number of patterns they share.

Besides that, non-orthogonal patterns also pose the first real challenge to the otherwise quite reliably working UP-state detection mechanism developed. Though the heuristic compensation proposed works adequately, an extension to the UP-state detection mechanism for non-orthogonal patterns could be developed.

Finally, retinotopic pattern completion as a particular application was investigated. It could be seen that as long as the encoded images were dissimilar enough, the activation of the corresponding

pattern was possible. However, if two images were very similar, the resulting small difference in stimulated MCs in the corresponding patterns is insufficient to activate one pattern over the other in a reliable way. A more systematic analysis is currently being performed. Another interesting approach would be the integration of the investigated network model into a full visual pathway with the aim of creating a very fast visual classifier.

Comparing the hardware-compatible model to the original model, the most prominent differences are the short dwell times (roughly half as long as in the original model) combined with too high average spike rates during UP-states (roughly twice as high as in the original model). Since the parameters in this model were fitted in essence by hand with only limited reference data (figure 3.2) available, it is likely that a more thorough fitting procedure with further reference data obtained from the original model will yield more accurate results. Because of neuronal adaptation, the maximum duration for effective layer 4 input is limited by the average dwell time – longer stimulus makes no sense because heightened adaptation constants will prohibit any further activity. Longer average dwell times would hence allow for longer periods of layer 4 stimulation which in turn could help to investigate retinotopic pattern completion, because stimulation times in which several patterns are stimulated at the same time can be increased.

Since all scripts in this thesis were developed with modularity in mind, all series of simulations conducted in this thesis can easily be regenerated, simulated and analyzed with new sets of parameters at practically no development cost. Furthermore, since simulation and analysis of single networks as well as evaluation and feature extraction of large sets of networks are handled by disjunct scripts which write all important derived data to disk, the range of features of the analysis suite can easily be extended. For example, the complete range of simulations for pattern completion and rivalry could be set up for every distorted network state examined in 3.3 *Hardware Imperfections*. The limiting factor is the simulation hardware available.

The main technical issue while carrying out this thesis which came up repeatedly was the fact that the earlier used *NeuroTools* as well as *PyNN* do not scale well when network size or simulation time are significantly increased. Especially the way of handling voltage traces and spike trains in *NeuroTools* finally led to its complete dismissal and emulation of needed functionality via custom routines to increase performance dramatically (see 2.3 *Problems*). All read and write operations of simulation data is *not* handled by *PyNN*. The code could be used by *PyNN/NeuroTools* to provide an alternate way of handling large sets of data that is only read once and then processed further – as was the case during this thesis (see 2.2.1 *Data Preprocessing*).

A large portion of the problems introduced by large networks or long simulation times could be somewhat mitigated by the introduction of a custom container file format designed specifically for storing voltage traces and spike trains. In the current state, voltage traces and spike trains are saved as simple text files which, when read again, need to be parsed and converted back to the corresponding data types. This is fine for small networks and short simulation times, but can get very tedious when the files in question are beyond 1 GB in size (which is, for instance, already the case for the spike data file when simulating a network with 6 HCs by 6 MCs for 300 s of biological time). Further improvements such as noting at which file position the different cells' signals are stored – a feature which is already present in the custom implementation of the needed *NeuroTools* functionality – could be stored directly at the beginning of the file when the file is created.

A Simulation Parameters

A.1 Neuron Model

Cell type	Used PyNN model
Pyramidal	EIF_cond_exp_isfa_ista
RSNP	EIF_cond_exp_isfa_ista
Basket	IF_cond_exp

Table A.1: Standardized PyNN models used,
see <http://neuralensemble.org/trac/PyNN/wiki/StandardModels>

Parameter	Pyramidal	RSNP	Basket	Unit
C_m	0.0692	0.007 69	0.007	nF
$E_{\text{rev},E}$	0.0	0.0	0.0	mV
$E_{\text{rev},I}$	-85.0	-85.0	-85.0	mV
I_{offset}	0.0	0.0	0.0	nA
τ_m	13.5	22.7	22.7	ms
τ_{refrac}	2.0	1.0	1.0	ms
$\tau_{\text{syn},E}$	28.0	28.0	6.0	ms
$\tau_{\text{syn},I}$	6.0	6.0	6.0	ms
V_{reset}	-75.0	-85.0	-80.0	mV
V_{rest}	-75.0	-75.0	-75.0	mV
a	0.0	0.0	-	nS
b	0.024	0.0012	-	nA
Δ_T	2.0	2.0	-	mV
τ_w	400.0	200.0	-	ms
V_{spike}	-57.0	-66.5	-	mV
V_{thresh}	-62.0	-71.5	-69.5	mV

Table A.2: Used neuron parameters in simulation

A.2 Synapses

Connection	Probability	Weight [nS]	U (TSO!)	τ_{rec} [ms] (TSO!)
Pyr \rightarrow Pyr (same MC)	0.25	1.10	0.25	575
Pyr \rightarrow Pyr (different MC)	0.30	0.19	0.25	575
Pyr \rightarrow RSNP	0.17	0.04	0.5	28
Pyr \rightarrow Basket	0.70	0.10	0.5	6
RSNP \rightarrow Pyr	0.70	3.80	1.0	4
Basket \rightarrow Pyr	0.70	3.00	1.0	4
Background \rightarrow Pyr	–	0.13	–	–
Layer 4 \rightarrow Pyr	–	1.10	–	–

Table A.3: Used parameters for neurons in simulation, weights fitted to default network size of 9 HCs by 8 MCs.

A.3 Scaling Formulae

Connection	Scaled connection probability \tilde{p}
Pyr \rightarrow Pyr (same MC)	$\tilde{p} = \frac{29}{N_{\text{Pyr}}-1} \cdot p$
Pyr \rightarrow Pyr (different MC)	$\tilde{p} = \frac{30}{N_{\text{HC}}} \cdot \frac{8}{N_{\text{Pyr}}-1} \cdot p$
Pyr \rightarrow RSNP	$\tilde{p} = \frac{8}{N_{\text{HC}}-1} \cdot \frac{30}{N_{\text{Pyr}}} \cdot \frac{7}{N_{\text{MC}}-1} \cdot p$
Pyr \rightarrow Basket	$\tilde{p} = \frac{30}{N_{\text{Pyr}}} \cdot p$
RSNP \rightarrow Pyr	$\tilde{p} = \frac{2}{N_{\text{RSNP}}} \cdot p$
Basket \rightarrow Pyr (Enlarging)	$\tilde{p} = \frac{1}{N_{\text{basket}}} \cdot p$
Basket \rightarrow Pyr (Shrinking)	$\tilde{p} = \frac{1}{N_{\text{basket}}} \cdot \frac{8}{N_{\text{MC}}} \cdot p$

Table A.4: Scaling rules for the connection densities p employed whenever a network deviated from the default size (for which each scaling factor amounts to 1). N_x represents the number of units of type x in the network to be scaled. p represents the original connection densities found in table A.3. Note that whenever a scaled probability \tilde{p} exceeded 1 it was kept at one and the weight of the corresponding synapses increased by $\tilde{w} = w \cdot \tilde{p}$.

B Further Information

B.1 Hardware Imperfections in a smaller Network

The following plots depict the same hardware imperfections as present in *3.3 Hardware Imperfections*, but with a smaller network of 6 HCs by 4 MCs which illustrates the size-dependence of hardware imperfections' influence on the network as well as the increased robustness of smaller networks.

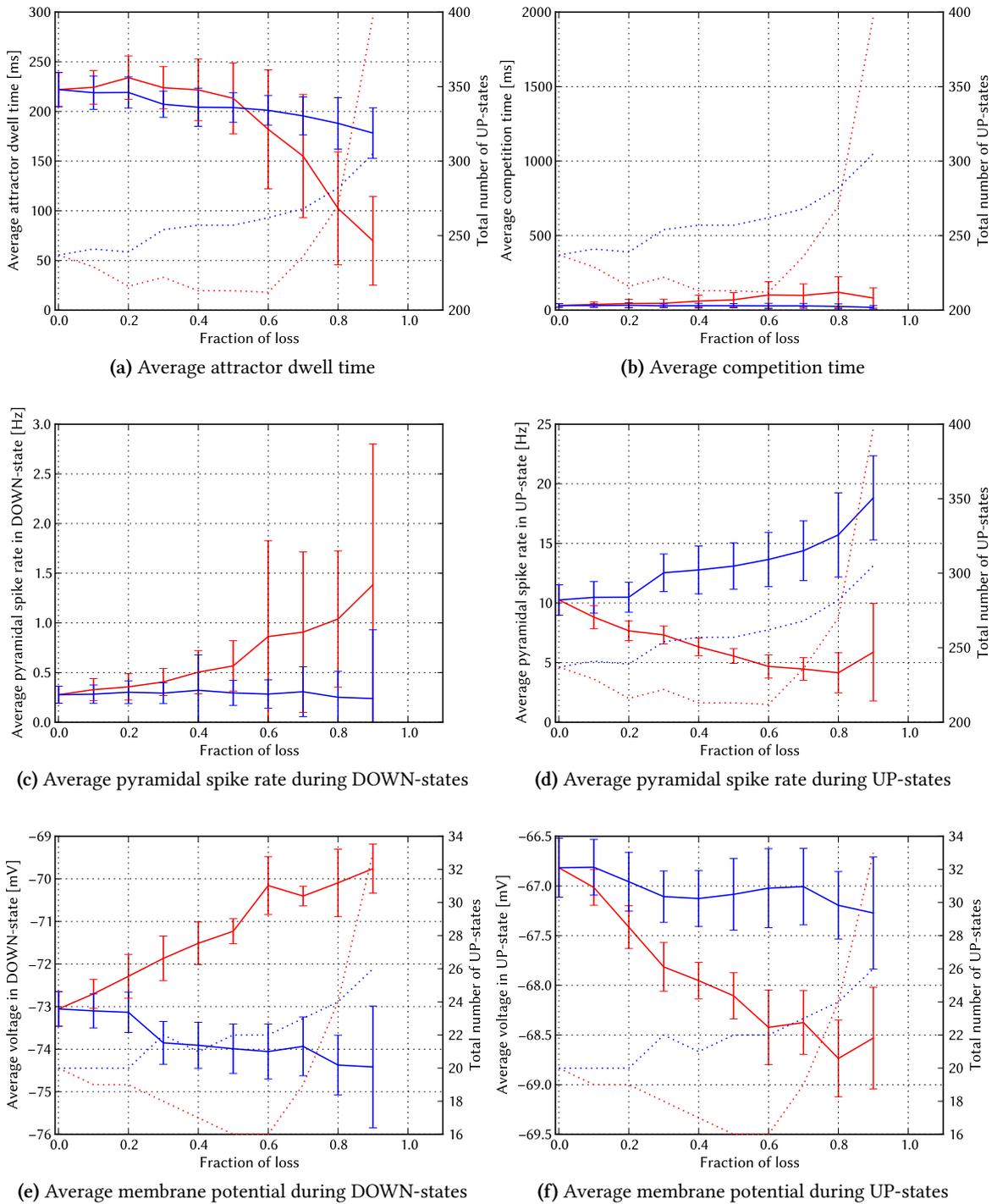


Figure B.1: Small sized network (6 HCs by 4 MCs) for varying degrees of pyramidal neuron loss. Neurons were lost homogeneously across all MCs. Both the uncompensated case (*red*, all other parameters are left untouched) as well as rescaling according to the rules in table A.4 (*blue*) are shown. In order to give a sense from how many data points the plotted values were computed, the total number of UP-states (*dotted line*) is plotted as well. Stimulation time was 60 s for (a) to (d) and 5 s for (e) and (f).

B.1 Hardware Imperfections in a smaller Network

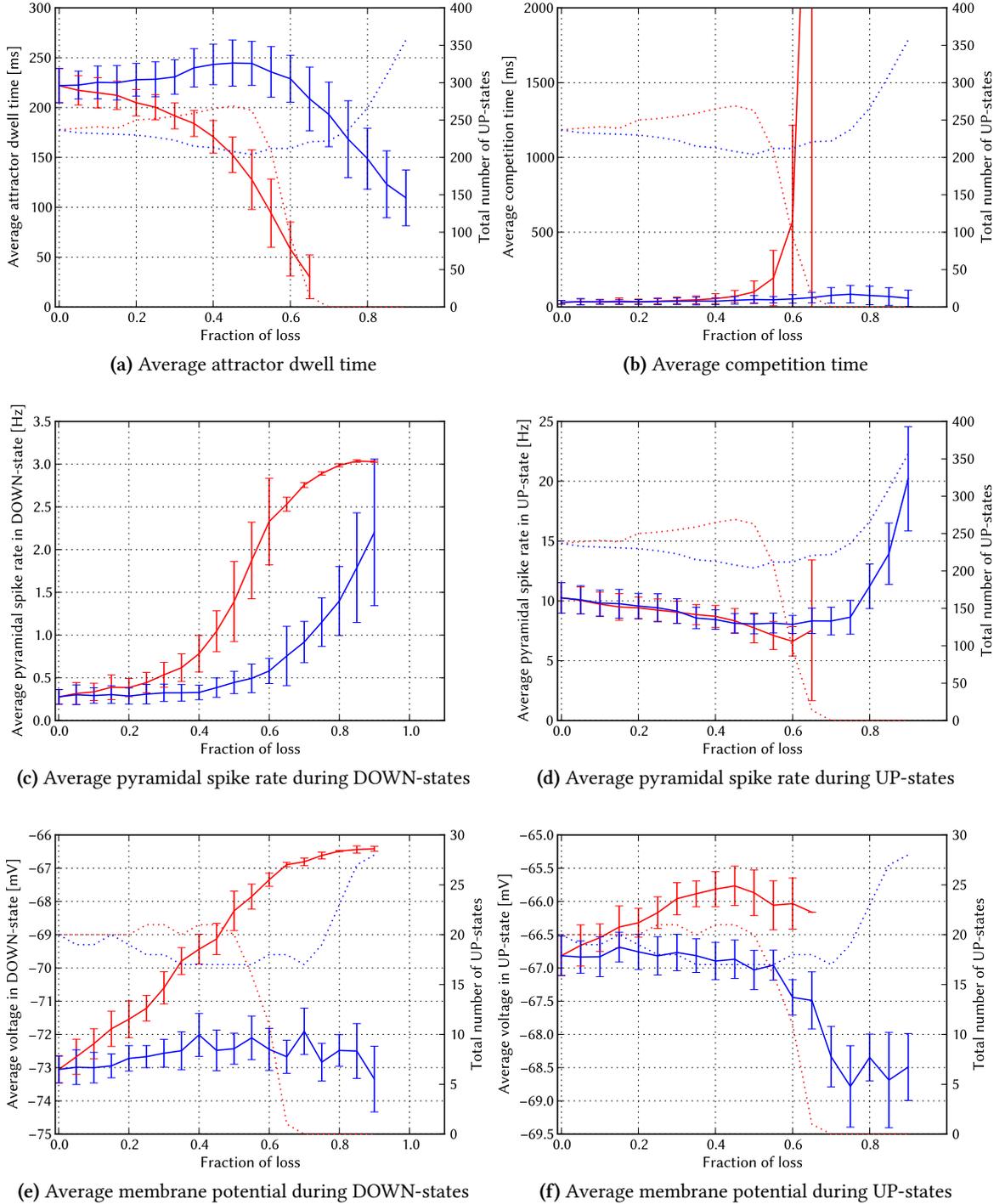


Figure B.2: Small sized network (6 HCs by 4 MCs) for varying degrees of total synapse loss. Synapses were lost homogeneously among all connections by reducing connection probabilities according to $\tilde{p} = (1 - f) \cdot p$ where f is the fraction of loss. Both the uncompensated case (*red*) as well as increasing the weights according to $\tilde{w} = w \cdot (1 - f)^{-1}$ (*blue*) are shown. The total number of UP-states (*dotted line*) is plotted as well. Stimulation time was 60 s for (a) to (d) and 5 s for (e) and (f).

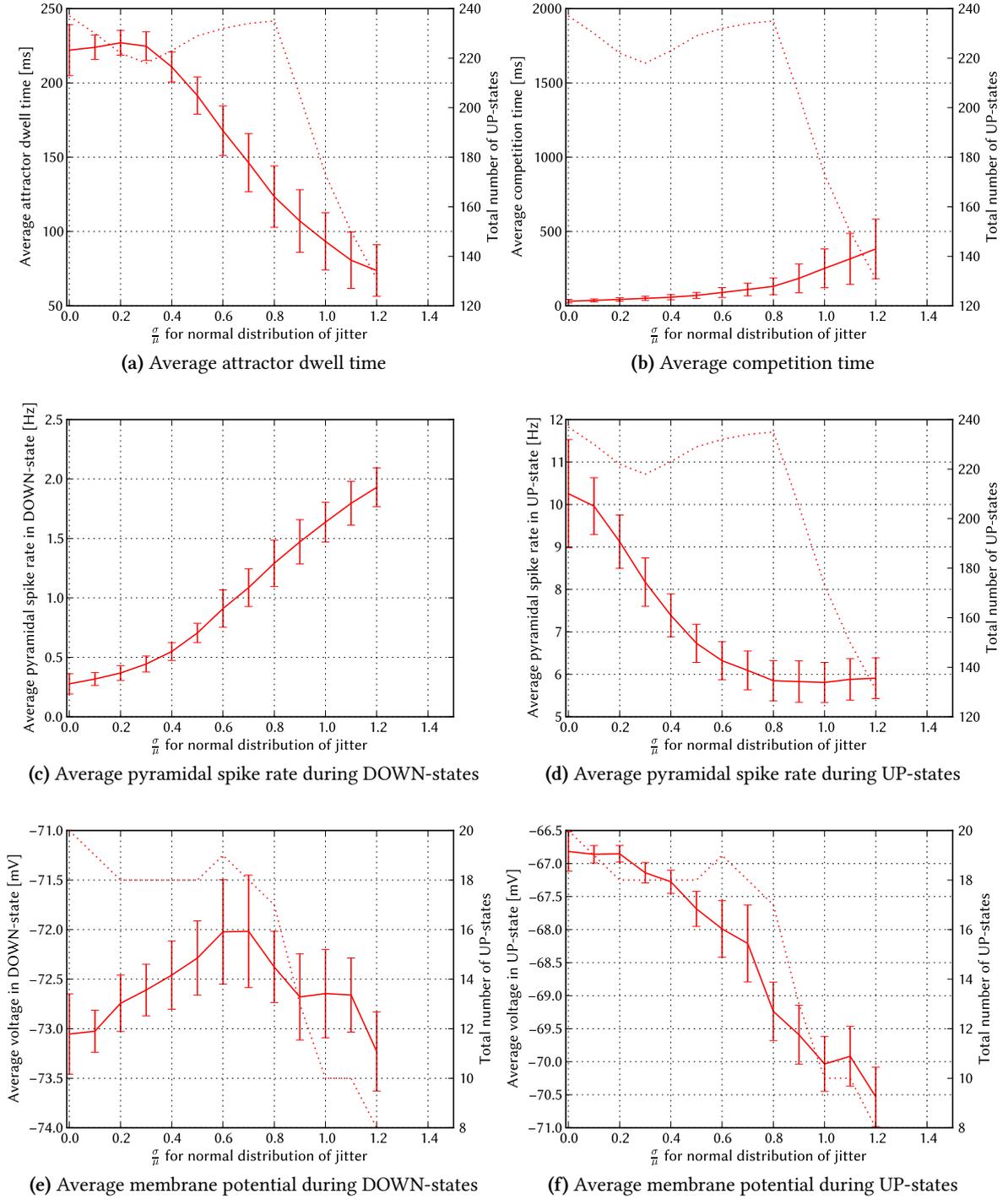


Figure B.3: Small sized network (6 HCs by 4 MCs) for varying degrees of synaptic jitter. Synaptic weights for *all* connections are distributed normally around their original values and varying widths σ . For each fraction of jitter σ/μ four different networks were simulated. In order to give a sense from how many data points the plotted values were computed, the total number of UP-states (*dotted line*) is plotted as well. Stimulation time was 60 s for (a) to (d) and 5 s for (e) and (f).

B.2 Diagonal Pattern Distribution Examples

Here are several examples of how non-orthogonal patterns may be distributed to allow a certain number of patterns per MC. The network possesses 4 HCs with 8 MCs each.

MC:	1	2	3	4	5	6	7	8
HC: 1	1, 9	2,10	3,11	4,12	5,13	6,14	7,15	8,16
2	1,16	2, 9	3,10	4,11	5,12	6,13	7,14	8,15
3	1,15	2,16	3, 9	4,10	5,11	6,12	7,13	8,14
4	1,14	2,15	3,16	4, 9	5,10	6,11	7,12	8,13

Table B.1: Diagonal patterns in a network of 4 HCs by 8 MCs: The patterns in which each MC participates are noted. Each MC participates in two patterns at once and each combination of patterns is unique.

MC:	1	2	3	4	5	6	7	8
HC: 1	1, 9,17	2,10,18	3,11,19	4,12,20	5,13,21	6,14,22	7,15,23	8,16,24
2	1,16,18	2, 9,19	3,10,20	4,11,21	5,12,22	6,13,23	7,14,24	8,15,17
3	1,15,19	2,16,20	3, 9,21	4,10,22	5,11,23	6,12,24	7,13,17	8,14,18
4	1,14,21	2,15,22	3,16,23	4, 9,24	5,10,17	6,11,18	7,12,19	8,13,20

Table B.2: Diagonal patterns in a network of 4 HCs by 8 MCs: The patterns in which each MC participates are noted. Each MC participates in three patterns at once and each combination is unique.

1	2	3	4	5	6	7	8
1, 9,17,25	2,10,18,26	3,11,19,27	4,12,20,28	5,13,21,29	6,14,22,30	7,15,23,31	8,16,24,32
1,16,18,31	2, 9,19,32	3,10,20,25	4,11,21,26	5,12,22,27	6,13,23,28	7,14,24,29	8,15,17,30
1,15,19,29	2,16,20,30	3, 9,21,31	4,10,22,32	5,11,23,25	6,12,24,26	7,13,17,27	8,14,18,28
1,14,21,27	2,15,22,28	3,16,23,29	4, 9,24,30	5,10,17,31	6,11,18,32	7,12,19,25	8,13,20,26

Table B.3: Diagonal patterns in a network of 4 HCs by 8 MCs: The patterns in which each MC participates are noted. Each MC participates in four patterns at once and each combination is unique.

C Acronyms and Technical Terms

API	Application Programming Interface	17
AdEx model	Adaptive Exponential integrate-and-fire model	5
BSS	BrainScaleS	2
FACETS	Fast Analog Computing with Emerging Transient States	2
HC	hypercolumn	3
LIF	Leaky integrate-and-fire	6
MC	minicolumn	3
PSP	Post-Synaptic Potential	21
PSTP	Phase Space Trajectory Projection	
RSNP	Regular Spiking Non-Pyramidal	3
STP	Short Term Plasticity	6
WTA	winner-take-all	4

D Bibliography

- Brette, R., and W. Gerstner, Adaptive exponential integrate-and-fire model as an effective description of neuronal activity, *J. Neurophysiol.*, 94, 3637 – 3642, doi:NA, 2005.
- Brüderle, D., et al., A comprehensive workflow for general-purpose neural modeling with highly configurable neuromorphic hardware systems, *Biological Cybernetics*, 104, 263–296, 2011.
- Cossart, R., D. Aronov, and R. Yuste, Attractor dynamics of network up states in the neocortex, *Nature*, 423, 238–283, 2003.
- Davison, A. P., D. Brüderle, J. Eppler, J. Kremkow, E. Muller, D. Pecevski, L. Perrinet, and P. Yger, PyNN: a common interface for neuronal network simulators, *Front. Neuroinform.*, 2(11), 2008.
- Hines, M. L., and N. T. Carnevale, *The NEURON Book*, Cambridge University Press, Cambridge, UK, 2006.
- Hunter, J. D., Matplotlib: A 2d graphics environment, *Computing In Science & Engineering*, 9(3), 90–95, 2007.
- Lundqvist, M., M. Rehn, M. Djurfeldt, and A. Lansner, Attractor dynamics in a modular network of neocortex, *Network:Computation in Neural Systems*, 17:3, 253–276, 2006.
- Marois, R., and J. Ivanoff, Capacity limits of information processing in the brain, *Trends in Cognitive Science*, 9(6), 296–305, 2005.
- Miller, K. D., Understanding layer 4 of the cortical circuit: A model based on cat v1, *Cerebral Cortex*, 13(1), 73–82, doi:10.1093/cercor/13.1.73, 2003.
- Millner, S., A. Grübl, K. Meier, J. Schemmel, and M.-O. Schwartz, A VLSI implementation of the adaptive exponential integrate-and-fire neuron model, in *Advances in Neural Information Processing Systems 23*, edited by J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, pp. 1642–1650, 2010.
- Müller, P., Distortions of neural network models induced by their emulation on neuromorphic hardware devices, Master's thesis, Universität Heidelberg, 2011.
- Oliphant, T. E., *Guide to NumPy*, Provo, UT, 2006.
- Petrovici, M. A., et al., A comprehensive analysis of causes, effects and compensation of limitations of the facets waferscale neuromorphic modeling platform (working title), unpublished, 2011.
- Shapiro, K., J. Raymond, and K. Arnell, Attention to visual pattern information produces the attentional blink in rapid serial visual presentation., *J Exp Psychol Hum Percept Perform.*, 20(2), 357–373, 1994.
- Shu, Y., A. Hasenstaub, and D. A. McCormick, Turning on and off recurrent balanced cortical activity, *Nature*, 419, 288, 2003.

- Tallon-Baudry, C., O. Bertrand, F. Peronnet, and J. Pernier, Induced g-band activity during the delay of a visual short-term memory task in humans, 1998.
- Thomson, A. M., D. C. West, Y. Wang, and A. P. Bannister, 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., *Cerebral cortex*, 12(9), 936–953, 2002.
- Tsodyks, M., and H. Markram, The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability, *Proceedings of the national academy of science USA*, 94, 719–723, 1997.
- Wandell, B. A., A. A. Brewer, and R. F. Dougherty, Visual field map clusters in human cortex., *Philos Trans R Soc Lond B Biol Sci*, 360(1456), 693–707, doi:10.1098/rstb.2005.1628, 2005.

Statement of Originality (Erklärung):

I certify that this thesis, and the research to which it refers, are the product of my own work. Any ideas or quotations from the work of other people, published or otherwise, are fully acknowledged in accordance with the standard referencing practices of the discipline.

Ich versichere, daß ich diese Arbeit selbständig verfaßt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, 31st August, 2011

.....
(signature)