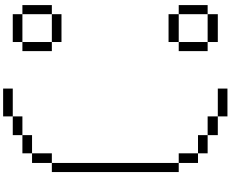
Synchrony Binding in Noise Excitable Coherence Resonance Networks

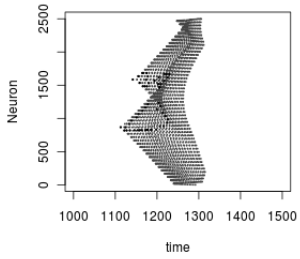
Jacob Huth

Osnabrück, December 12, 2011

Abstract

Using a noise excitable media FHN network we investigate how synchrony binding in activity patterns can be created and analysed. As a goal a spatial pattern (such as the smiley below) should be recoverable from the relative spike time delays.





Foreword

This thesis stands in context of a collaboration with Clemens Korndörfer, Ekkehard Ullner and Gordon Pipa.

Contents

Chapter 1

Introduction

Neurons communicate information to other regions of the brain through spikes, or action potentials. In addition to the believe that the number of spikes carries important information, theoretical thoughts about the speed of computation in eg. the visual system[Singer, 1993], as well as physiological findings suggest that their relative spiking delays in the range of very few, or up to fractions of milliseconds, are even more important.

Propagation of spikes, as well as synaptic plasticity are more sensitive to co-occurring spikes, precise timing is therefore a lot more beneficial for learning and signal transmission than a simple rate code. But another, very beneficial effect of precise spike time processing is that a homogeneous population of computational units can integrate their computations about the "same thing" without changing their synaptic connectivity. It is suspected [Gray et al., 1989, Engel et al., 1991, König et al., 1995], that those neurons in the visual cortex, that observe something that is perceived as belonging to the same object, fire in synchrony, instead of eg. using a multiplexer circuit and separating its communication by mechanical routing. The signal is separated in time, so that subsequent processing areas can learn that the synapses transmitting it are all causally connected to the stimulus. This is called **binding** by synchrony and is thought to be the neural correlate to perceptual grouping (eg. Gestalt laws). [Singer and Gray, 1995]

The mechanisms for this process is either a *top-down* driven process, in that a higher cortical area recognizes an object and sends feedback to synchronize the lower areas, a *bottom-up* process that generates synchrony out of gradually changing features and horizontal connections or even a non-hierarchical process that simultaneously integrates top-down and bottom-up computation.

Evidence for bottom-up binding via temporal features could be that object segregation works best on temporally dissonant input [Leonards et al., 1996]. Delays in the input as low as 16ms can lead to the perception of two distinct objects. But still cross-hemisphere 0-phase synchrony in cat visual cortex disappears with a cut of the corpus callosum[Engel et al., 1991], suggesting that rather than being synchronous due to having a common stimulus onset, the two

regions seem to also mediate each other into synchrony.[1]

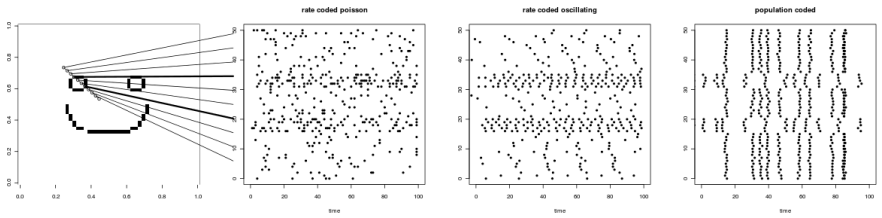


Figure 1.0.1: Temporal codes of a simple stimulus: a Poisson rate code, a regularly firing rate code and a synchrony population code with equal rate. It is possible to identify the neurons corresponding to the eye contour in each case by either counting spikes or comparing spike times. If speed is important for processing, the Poisson rate code would be uninformative for most short time frames, while the other methods provide good estimates.

1.1 Happy Cortex - an Overview

In our simulation, we assume a pure bottom-up process: subsequent areas can only observe the activity, but not influence it. Also we abstract the actual complexity of the cortex into a model of *noise excitable media*, which reflects in its connectivity the spatial structure of observed figures. We assume those connections to be predetermined and fixed neighbourhood relationships, rather than a system of complex self organization and connectivity along multiple feature dimensions.

We hope with this simple construct to suggest the possibility of pure bottom-up figure binding and explain how one such process can actually lead to synchrony and what limits apply to the model used. Ullner et al.[Ullner et al., 2008] have already shown that for connected lines or loops of neurons, the population can be brought close to 0-lag synchrony, due to stochastic properties of the oscillators. Building on this finding, we try to extend the 1-dimensional into a 2-dimensional case and explore how the neuron population can be divided into a synchronous and a phase-orthogonal or asynchronous population.

To do this we explain in Section 1.2 the idea of **Coherence Resonance**, which is applied in Section 1.5 to an FHN model with fixpoint dynamics. The most prominent activity pattern that arises when these models are coupled and a small section is excited are **waves** (Section 3.2). These waves can create synchrony populations, as demonstrated in Section 3.3 and explained in 4.1.

In Section 3.4 we then try to create distinct synchrony populations, measured with tools established in the *Methods* Chapter. Using a simplified model of

[1] Whether the corpus callosum can be considered a hierarchically higher brain area would be debatable.

waves as uniform radial phase changes (2.4), we conclude about the connective-geometric properties of stimuli and their relation to synchrony population separability.

1.2 Coherence Resonance

Coherence Resonance is an effect of oscillatory activity being evoked and enhanced by noise. A number of dynamical systems such as optical feedback systems [Giacomelli et al., 2000], but also climate systems and chemical reactions, reflect a stochastic property of their input with enhanced or reduced regularity in their activity. In most cases this will be a peaked distribution with a maximal coherence - that is most regular oscillation - for a certain level of variance in the input and reduced regularity when increasing or reducing the variance. This is because a variation is needed to excite the system, but too much variation disturbs its state trajectory. The exact properties of the model used by us is discussed in Section 1.5.1.

Using noise as input allows us to formulate spikes in a stochastic process without adding other noise sources, further we can claim that the temporal output of the structure would in subsequent processing steps refine the signal that the oscillators are sensitive to (see Figure A.3.1 in the appendix).

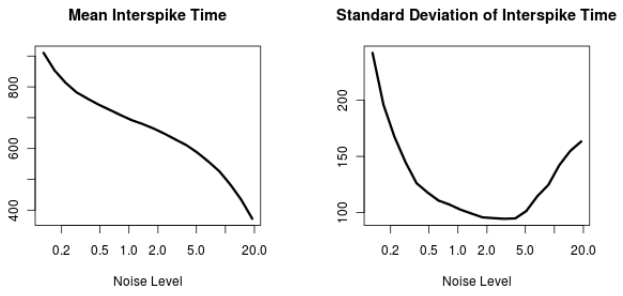


Figure 1.2.1: Demonstration of optimal coherence in FHN neurons as produced by "cr" simulation. While the mean firing time steadily decreases, the standard deviation of how long the period between two spikes are has a minimum. The measure of noise is explained in Section 1.5.2

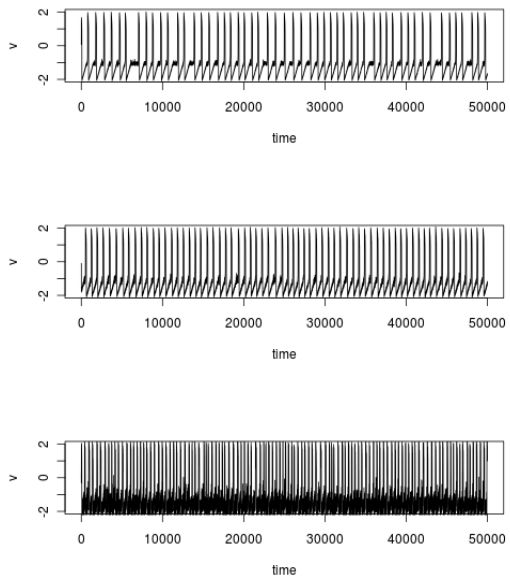


Figure 1.2.2: Three different noise levels: low, optimal and strong with mean and standard deviations of firing time as 853±242, 725±110 and 372±163 ms respectively.

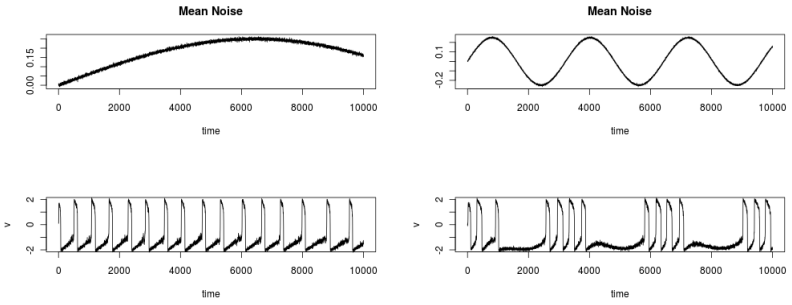


Figure 1.2.3: Low frequency input modulation, simulated here by superimposing a slow sinusoid input onto the membrane potential, does not influence the spiking behaviour up to a certain frequency threshold. Here we see a slow modulation (left), that does not influence the behaviour at all, and a faster one (right) with equal amplitude (from -0.2 to +0.2) that inhibits spiking at certain times and produces fast spike bursts in accordance to its derivative: on up slopes we have bursts, on down slopes silence.

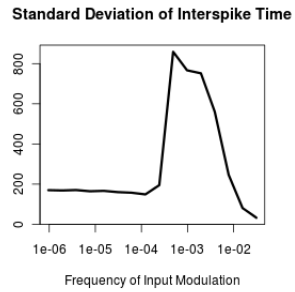


Figure 1.2.4: The Standard Deviation of Inter-Spike-Time for varying frequencies of input modulation with the sudden rise after 0.0001 being explainable by Figure 1.2.3.

1.3 Waves

Waves are solutions to the coupled differential equations of oscillators (or *operating regimes*) in which each oscillator has the same frequency and the phase is spatially coherently distributed. Some of our noise excitable oscillators will be excited by input, in this sense they are spontaneously activated. They will start to excite their neighbours, which will excite further oscillators. This process relies on two time scales: how fast the next point can be excited and how fast each point will fall back to its resting point. If the transmission is faster then the reset, the wave will continue to propagate, otherwise it will die out. Since we only count the spikes and do not have a special interest in their shape it is sufficient to know whether the amount of time that the neuron is excited is enough to excite it's neighbours relies on the strength of synaptic coupling and that we can choose appropriate couplings to encourage propagating behaviour.

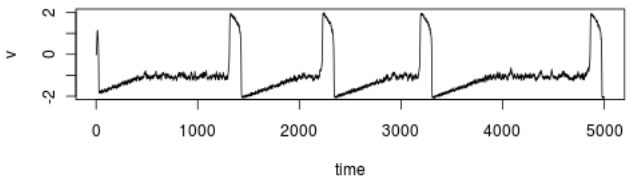


Figure 1.3.1: Membrane potential of a FHN neuron over time with sufficient input to cause spikes

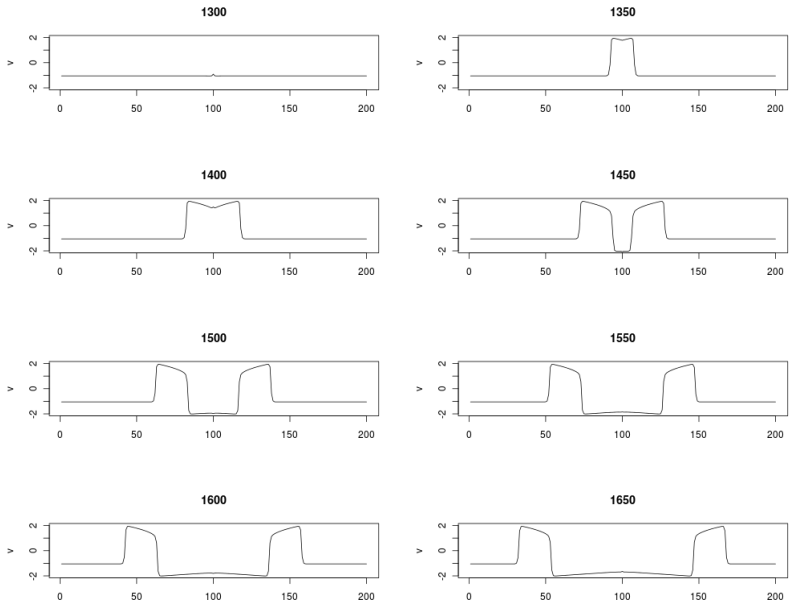


Figure 1.3.2: A wave seen across neurons at different times. The shape mimics that of an individual action potential since these look identical for each neuron. The time delayed transmission activates the units in linear order, hence each units phase in the wave is at each instance between the two phases of its neighbouring neurons forming a continuous activation wave. This setup is also a model for action potential transmission inside an axon, rather than between neurons.

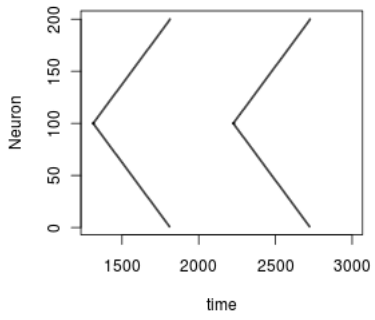


Figure 1.3.3: The same wave as seen in Figure 1.3.2 as a raster plot. The apparent lines are actually successive spikes.

1.4 Spatial Structure

We use a 2d grid to give our network more freedom in how the activity can spread, while still having the opportunity to inspect the whole network state as a square lattice in which activation will spread spatially coherent and cause and effect can be easily interpreted.
Taking the idea of [Ullner et al., 2008] that closed and open contours synchronize themselves through coherence resonance, the question arises if this synchronization actually segments the activated from the not-activated neurons if they are connected in a regular fashion, without special knowledge of the stimulus.
The neurons in the chain or loop setup of [Ullner et al., 2008] are connected only as direct neighbours. Embedding the contour in a background population therefore requires that this relationship is preserved and also that the population is connected homogeneously without special knowledge of the contour that is to be segmented. This leads in its simplest realization to a 2d grid with strong neighbour-neighbour connections either in a 4-edge or 8-edge lattice as seen in Figure 1.5.3. A similar setup was used in [Perc, 2005], who also observed waves, spirals and synchrony in a 2d noise excitable media.

1.5 General Simulation Setup

The simulations are run in GNU R [R Development Core Team, 2011] and the source code is available as additional material (see Appendix). During the simulation, only spike times are recorded (with the exception of simulations that ought to show dynamics in the membrane potential) and later visualized and various measures computed from them.

1.5.1 Model

The Fitz Hugh Nagumo model is a simplification of the biophysically inspired Hodgkin-Huxley Model [Izhikevich and FitzHugh, 2006]. It consists of two state variables, one being the observable membrane potential, the other a slower recovery variable - an umbrella term for all inhibiting channel variables of the Hudgkin-Huxley model. We are using a version in the noise excitable regime with vertical u-nullcline, making it a "Class 3" Neuron[2] - being only excitable by strong pulses rather than a slow ramp current; there is no rheobase since the model adapts easily to voltage changes. That means for connected neurons that they can excite the neuron with their own action potential, which is itself a very fast increase in membrane potential, but for the input stimulus, that similarly fast deviations are needed.

[2]HH Class 3 neurons are explained in Section 7.1.4 in [Izhikevich, 2007]

$$\dot{v} = \frac{1}{\varepsilon}\left(\frac{v^3}{3} - u + I_t\right) \tag{1.1}$$

$$\dot{u} = a + v \tag{1.2}$$

$$I_{n,t} = Noise_{n,t} + \sum_{n_i} S_{n,n_i} * (v_{n_i,t-1} - v_{n,t-1}) \tag{1.3}$$

In this case we chose a straight u nullcline instead of a tilted one as can be seen in the phase portrait. This leads the model to adapt to slow voltage changes but respond to rapid ones. With $a = 1.05$ or $a = -1.05$ we have a stable fixpoint attractor that will lead the neuron to a resting state. These two parameters

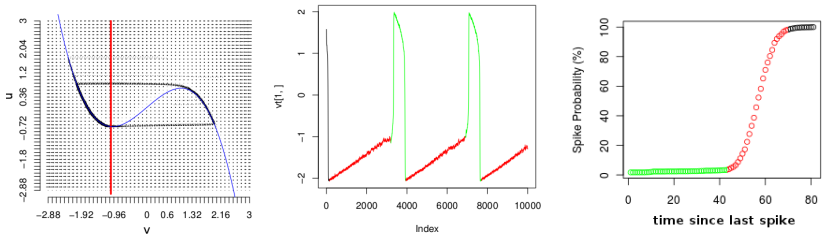


Figure 1.5.1: Phase portrait and spike shape of the FHN neuron. On the right side of the straight red u-nullcline the u-movement is slightly upwards, left of it downwards and a rapid v-direction rightwards below the cubic blue v-nullcline and leftwards above resulting in a limit cycle over both "knees" of the v-nullcline. The rightmost plot shows the interspike time probability, separated into the two phases of spike trajectory (green) and re-excitation time (red). While in the trajectory, the probability for a spike is very low, the re-excitation time is normal distributed. The width of the distribution (slope of the sigmoid) determines how stable the frequency of the oscillator is.

make the model noise excitable, meaning that it will only spike if sufficiently fast noise pulses (eg. pre-synaptic input) are applied, otherwise it remains at the fixpoint.

If we compare the noise excitability to the regular spiking version of the model (eg. $a \approx 0.05$), we can clearly see the difference in behaviour (Figure 1.5.2). Although the regular spiking neuron is not disturbed by the noise until it reaches a critical value, the noise excitable system does not fire at all for very small noises, spikes regularly for noises with standard derivative of over 0.2.
Also note that the maximal frequency is lower than the frequency of the regular firing model.
Spikes are counted as the change of sign of membrane potential from negative to positive.

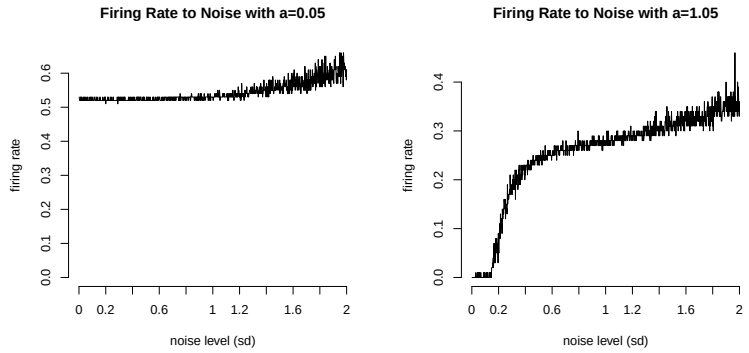


Figure 1.5.2: The difference of FHN Neurons in "regular firing" and "noise excitable" mode. While the firing rate of the neuron with a close to 0 is not influenced by the noise level, the noise excitable model requires a certain amount of noise to fire at all.

1.5.2 Input

Since we use noise excitable oscillators, the input they receive is white noise that can be adjusted in its amplitude, ie. its standard deviation. In Figure 1.2.1 the noise level was already used to demonstrate the effects of coherence resonance, with noise varying with a sigma from 0 to 20. The noise is generated by the R random number generator as a normal distribution with parameters $mean = 0$ and $sd = \sqrt{sigma}$. [3] Through coherence resonance the probability of spiking (and therefore firing rate) of a FHN neuron as used here, is monotonic dependent on $sigma$. The noisier the input is, the more the neurons will spike.

In two population experiments, neurons are segmented into two populations receiving independent input. One population - called the "image" - receives stronger noise, while the "background" population receives only a fraction of the noise level or no varying input at all. For example a square of 10×10 neurons could receive an input of $sigma = 1.2$ while the background receives $sigma = 0$.

1.5.3 Network

As stated before, the network structure is a regular grid with uniform connections between each pair of spatial neighbours (see Figure 1.5.3). The borders are connected to a torus to eliminate spatial inhomogeneity, so activity travelling out the right side will reappear at the left side. We can interpret the "image" population as a spatial two dimensional binary picture with properties such as area, perimeter and connectedness.

[3] using the standard "Mersenne-Twister" random number generator [R Development Core Team, 2011]

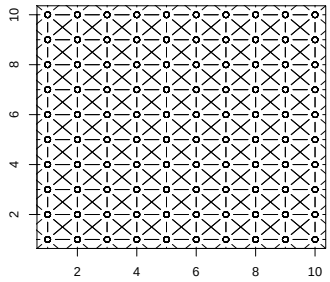


Figure 1.5.3: The synaptic connections between the oscillators in an 8-connected lattice. Each neuron is connected to its eight neighbours.

Diffusive Coupling

Diffusive coupling (see Equation 1.3) or gap coupling can be understood as a mutual pulling force, like a rubber band, that tries to keep the state of two coupled systems together. Biologically a connection between two cells, piercing each membrane, causes the ion concentration of both cells to assimilate. The further the concentrations are apart, the stronger the urge for ions to switch sides. Mathematically this means that we add to the membrane potential of each unit the scaled difference to each connected unit. This can either excite it by dragging its state over the separatrix, or have a dampening effect if the v-derivative of the spike onset is not strong enough to overcome the force applied by the surrounding units. In Figure 1.2.3 we can see how a rapid change in input can speed up or silence the neuron, which is also what our coupling does to excite neighbouring units.

Chapter 2

Methods

2.1 Output of Simulation

Looking at the raster spike plot for any of the simulated conditions, we see something like Figure 2.1.1.

Since the neurons are two dimensionally connected, but we have only a one

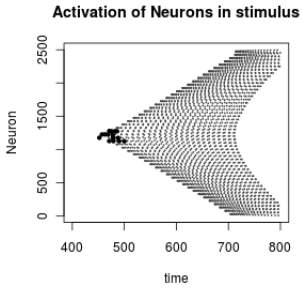


Figure 2.1.1: A rasterplot showing the activity of the network. Each line corresponds to one neuron, each dot is a spike of said neuron. Darker dots are spikes of image neurons.

dimensional way of listing them in this plot, one must keep in mind, that the activity skips over 1 row to its horizontal neighbours, but 50 rows to its vertical neighbours, hence the apparent interference pattern for travelling activity.
In this plot the activity actually spreads out from the center to the surround where it collides with itself (since our network is cyclic) or the background is excited enough to activate spontaneously. The animation over time would evolve in some fashion like Figure A.1.1 on page 32.

The temporal delay that is evident as the "speed" of the wave depends on the time constants of synaptic coupling and excitation dynamics with a maximal

speed of one "hop" per timestep for instant spike propagation.

2.2 Defining Synchrony Binding

If we want to define synchrony binding in terms that we can measure, we can orient ourselves to the physiological findings, which claimed to see it in the brain, and therefore had to have a method to measure it. The solution developed by Knig et al. was to compute a correlogram between spike trains or mean field potentials, and measure the peak height at 0, as well as the periodicity of the signal for evidence of oscillation.

The oscillation characteristics can also be assessed in Inter Spike Interval Histograms, that show how much the frequency varies. In Section 1.5.1 we used a cumulative ISI Histogram to show how the probability for a spike increases sharply in a narrow region to produce regular oscillations. With this method (and as a scalar the standard deviation of the ISIs) we can assess the regularity of individual oscillators, and also compare their frequency, but to find synchronous populations, we need to take the phase into account as well.

To have an overall estimate on the general synchrony in a population of neurons, the variance of the mean field seems to be important as uncorrelated spikes lead to a relatively constant signal, but simultaneous spikes have a higher amplitude at the time of spikes, but a lower one before and after. If we want a comparable value, it is wise to divide the variance by the mean of the individual variances, as this will be bounded in [0,1]. This measure is called R_{syn} and used by Ullner et. al. It is discussed in comparison to the Tempotron in the Appendix under A.4.

$$R_{syn}(x) = \frac{var_t(mean_n(x))}{mean_n(var_t(x))} \tag{2.1}$$

If we want to compare a small number of spike trains with R_{syn}, we have to convolve them with a large kernel first, as the variance of the mean field will treat close-to-synchrony with the same measurement as complete asynchrony. To actually find out which neurons are bound (and therefore if the binding process works), we either have to compare every spike train to every other, resulting in a scalar value of synchrony, or we have to divide the global population into arbitrary subunits, each of which have an R_{syn} value. While the later method yields acceptable results if we have suspicions about which populations have to be inspected (eg. the stimulus vs. non-stimulus or each local neighbourhood), to find the best-bound population, we would have to inspect the whole power set, which is even less desirable than the $O(n^2)$ complexity class of comparing pairs. Since we can not use mean field analysis on two spike trains without having to make arbitrary choices in the dynamics of a kernel used to convolve the spikes, we might look to a mechanism that biologically is supposed to benefit from synchrony, which is Spike Time Dependent Plasticity (STDP). In general synapses strengthen eg. if the post synaptic cell spikes a very short time after the pre synaptic cell(s), assuming a causal relationship between the spikes and

so reducing redundancy and benefiting signal speed. If we then imagine a subsequent layer of neurons with STDP synapses to every neuron in our grid, we would assume that after a number of spike waves, the synapses would favour the synchrony bound neurons. Instead of building an actual model for this, we only use the idea of the short time frame, a threshold of spike time difference we use to determine which Neurons are synchronous. But since this threshold can be quite arbitrary, it is useful to think about whether we can formulate a threshold like this in general terms and fit it to each simulation.

To see if a threshold could separate neurons into synchronous and asynchronous populations, we have to take a look at how the distribution of "closeness" can be analysed. To this end we define the *mean minimal spike distance*:

$$distance(n_1, n_2) = \underset{i \in sp(n_1)}{mean} \left(\underset{j \in sp(n_2)}{minimum}(\mid i - j \mid) \right) \tag{2.2}$$

(with $sp(n)$ the set of spike times (in ms) from neuron n)

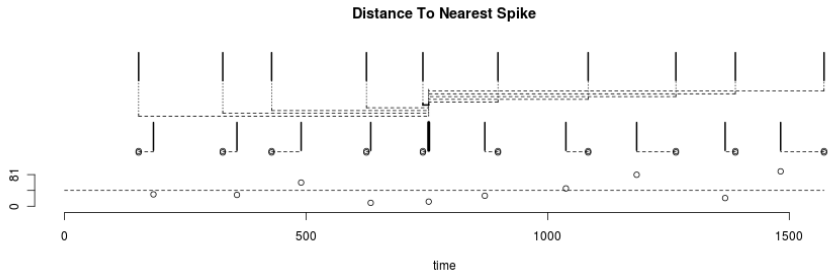


Figure 2.2.1: A graphical representation of how spike distance is computed. For each spike (eg. the bold spike in the lower spiketrain) the distance to the temporally closest spike is computed. The mean over those values for every spike in the reference spike train (the dashed line in the lower part of the plot) can be used to compare the similarity with other pairs of spiketrains. When the time between waves is larger than the largest distances within the wave, the measure is symmetric.

If we use a spike distance measure as defined in Equation 2.2, computing the distance from a fixed neuron (preferably one from the stimulus population) to every other neuron is fairly uncomplicated and yields us a distribution of spike distances. Looking at this distribution from one of the simulations in Figure 2.2.2 and 2.2.3, we can already suspect that linear classification with a threshold might be possible, but will result in a number of false positives and false negatives. We might assume that either a few neurons in the stimulus did not synchronize and we therefore would not want to include them in the bound population anyway, or that because of the stochastic nature of the spiking mechanism, we were just unlucky with our choice of reference neuron. In principle it could be possible to combine measurements from multiple reference neurons to

extract a synchrony bound population, but right now this is not more than an interesting problem. To estimate the goodness of binding we can compare the extracted ($d <$ threshold) population to the actual stimulus, either by number (false positive/negative errors) or by spatial fit. Alternatively we can quantify the separability of the two assumed populations by comparing their mean values. The closer they are, the more false positives a threshold and therefore STDP would generate. Assuming that we actually succeeded in separating the stimulus from the background, we can simply calculate the ratio of mean distance to some reference neuron, of each stimulus respectively non-stimulus neuron. If our reference neuron is drawn from the stimulus population, we would expect a value less than one, if it was from the background population, a value greater than one instead.

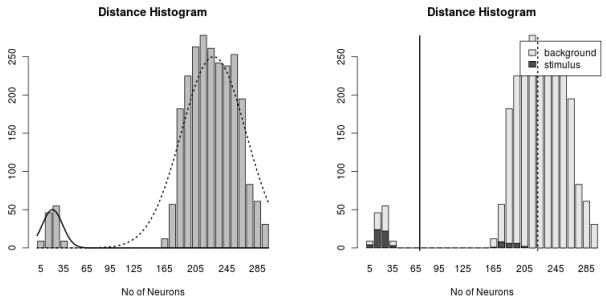


Figure 2.2.2: A separable Distribution of Spike Distances: The first plot assumes two distinct distributions that are easily separable, yet if we include the information about which neurons are actually in the stimulus, we see that a few of them are bound incorrectly. Still the mean of stimulus and background population show that a linear separation can identify at least most background neurons.

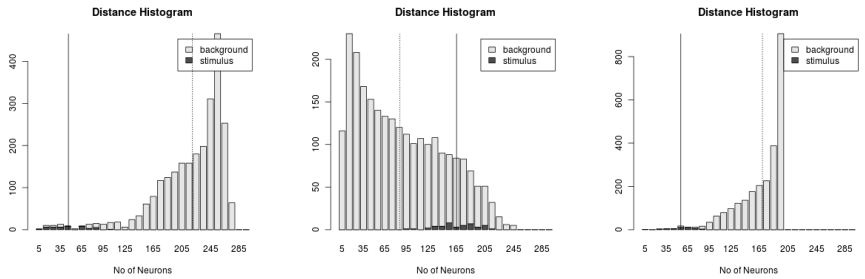


Figure 2.2.3: A not clearly separable Distribution of Spike Distances: Only with knowledge about the stimulus can we assess if the distributions of this simulation have different means. To extract a synchrony population, a combination of distance values into a linear classifier, or a simple threshold for the third reference neuron are possible. The vertical lines denote the mean of stimulus and background population, so the ratio would be smaller 1 for the first and third, and greater 1 for the second reference neuron, which actually fits to which population they were drawn from.

2.3 Distance of Spikes

Using the relationship defined in Equation 2.2 (also see Figure 2.2.1), we can compute heatmaps from one neuron to every other in the population. If we managed to synchronized a population, the mean spike distance within should be lower then from within to outside. We can suspect that spatially close regions should be closer spike distance wise than spatially removed regions, since they are connected with fewer interneurons.

For the activity in Figure 2.1.1 on page 14 these heatmaps show for each neuron a region of minimal distance radially around the center (see Figure A.1.3 on page 33), as should be expected since those are the regions that are active one at a time. We can find the largest distances to the surrounding neurons, if we pick the neuron that fired first. It will highlight other spontaneously firing neurons as well as early wave onsets.

Also noteworthy about the mean minimal spike distance is that:

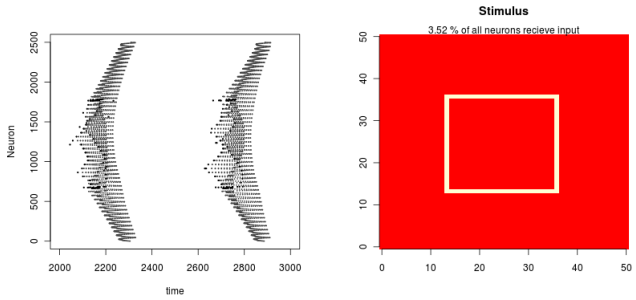
- The measure is not bound to be symmetric: a neuron with only one spike will have a distance of maximally f/2 to a neuron with firing frequency f, while this neuron will have a distance of T/4 to the other neuron. We can only assume symmetry if the frequencies are equal and the distance between waves is larger than the distances within the wave.

- In repeating waves the maximal distance (first to last spike) gives us the width of the wave and by this another measure of overall synchrony

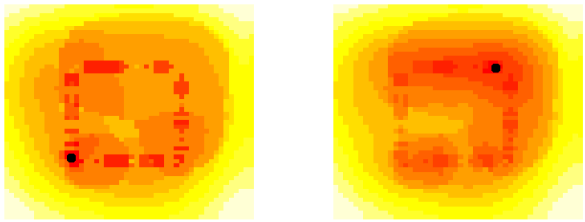
- We can try to assume near-transitivity in cases of symmetry, that is - given symmetry conditions - if a is close to b and b close to c, a might be

close to c.

- Computing the variance instead of mean can estimate non-0-phase-locking behaviour.



In stimuls neurons: 664 and 1784



Out of stimulus neurons: 1250 and 1225

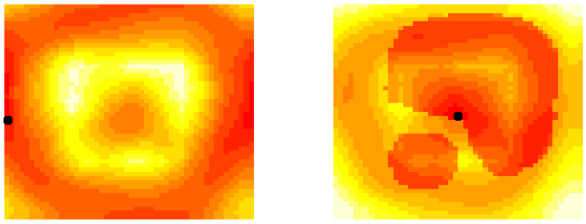


Figure 2.3.1: Four examples of spike time distance heat maps from Neurons 664 (lower left corner of stimulus), 1783 (top right corner of stimulus), 1250 (midway on the right border) and 1225 (center). The neurons inside the stimulus have a low distance to other in-stimulus neurons, those outside have a high distance to in-stimulus neurons. Another example is Figure A.1.3 on page 33

2.4 Simplified Explanatory Model

In Section 3.2 we will discover that in a mostly unexcited, but highly coupled grid, a few excitable units suffice to create global activity by emitting radial wave-like activity that will be used in 4.1 to explain the mechanisms that can be used for binding. Neglecting the modulating effect of noise on wave transmission speed, we can assess how many spikes have to be synchronized to convince the spike distance measure that a synchrony population has been established. For this we take a random subset of stimulus neurons and calculate the phases of other oscillators that are activated by waves by assuming that the phase will be equivalent to the distance:[1]

$$phase_{t,x,y} = \sqrt{min_i((x(s_{t,i}) - x)^2 + (y(s_{t,i}) - y)^2)} \tag{2.3}$$

$$distance_{x_1,y_1;x_2,y_2} = \langle |phase_{t,x_1,y_1} - phase_{t,x_2,y_2}| \rangle_t \tag{2.4}$$

In Section 3.5 we compare the average outcome of analysing any number of stimulus neurons in any number of waves. All stimulus neurons activating simultaneously should produce the clearest token of synchrony binding, while less sources result in less binding.

[1] With the functions x() and y() providing the coordinates of the spiking neuron in x and y dimensions, t denotes the wave number and $\langle \ldots \rangle_t$ the average across waves.

Chapter 3

Results

Interpreting the spike times generated by the simulation with the proposed methods was done part qualitative- and part quantitatively. We started with an uncoupled population to assess the coherence resonance characteristics and then used a coupled but uniformly stimulated population to find optimal noise and connectivity parameters for wave and synchronization behaviours. With those parameters we then restricted the CR optimal noise parameters to a stimulus population, while the background received only minimal or no input.

3.1 Low coupled dynamics

Reducing the coupling to 0 will have each oscillator acting as an optimal noise detector, which makes it possible to bind "distant"[1] neurons into loose synchrony by giving both optimal noise with an equal onset, resulting in a almost stable firing frequency which due to synchronous input onset is roughly in the same phase as seen in the mean field in Figure 3.1.1 on the next page. Since the deviations accumulate over time, the synchrony will fade some time after stimulus onset.

The phase difference is dependent on the excitability mechanics of the neuron. Looking at Figure 1.5.1, one plot shows the sigmoid of spike probability some time t after the last spike. The steeper this sigmoid, ie. the more reliable the neuron will spike at a specific time t, the closer phases of an uncoupled population will be. This sigmoid depends on the a parameter of the model, but more importantly on the noise level, which, by increasing the occurrence of excitable stimuli, squeezes the distribution of re-excitation time, increasing firing rate and regularity.

Using small quantities of coupling that act as a modulator for noise sensitivity (see Figure 1.2.3), the coherence across the population will increase as the small deviations in phase can be nullified by neighbour excitation.

[1] distance being infinite without coupling

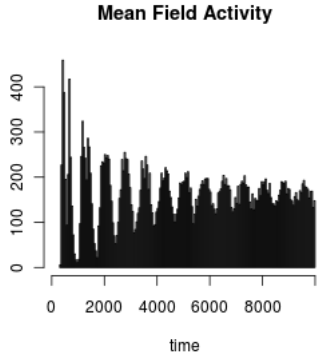


Figure 3.1.1: Loose synchrony binding due to synchronous onset and equal frequency. After a few spike waves the synchrony washes out due to instabilities in the spiking trajectories.

Populations with different input noise levels are also distinguished with different rate codes (see Figure 1.5.2). This property is lost when coupling is strengthened since the firing rate is equalized across the whole population when waves are formed.

3.2 Wave dynamics

Through diffusive coupling a spike of a neighbouring neuron provides enough drive to a neuron to cross its separatrix and go into a spike trajectory itself. The delay from each neuron to its neighbour generates a gradually changing phase over the whole space. In Figure A.1.2 and A.1.3 we see such activity and the corresponding distance analysis. A wave produces through the spatially changing phase synchrony populations radially around the source. Each neuron is synchronized to all those neurons that have the same distance to the source, plus/minus the effects of noise that can speed up and slow down the wave transmission stochastically.

To better analyse wave dynamics, we can restrict the source of the wave spatially to a small region that receives input or alternatively keep the noise level at a rate that makes it unlikely that more than one neuron creates a wave at any time. In the spatially fixed version the oscillation pattern and synchrony populations will be constant over time and the size of each synchronous population will be determined by the distance to the source[2], but if the source is

[2]The further we are away from the source, the more neurons have this distance - ignoring boundary conditions where the number decreases again - called a *Square Point Picking* and *Square Line Picking* problem, but with the additional constraint that we are on a torus. Instead of deriving the actual distributions it is sufficient for our cases to estimate the *big O notation* by numerical experiments.

randomly determined by CR, the synchrony will even out over time as the mean distance converges to $\frac{1}{2}\sqrt{N}$.

If we then add more noise, so that eg. two neurons in each wave have the opportunity to spike, the distances will decrease as the mean of randomly drawn distances scale with $\frac{1}{\sqrt{N_s}}$ (N_s the number of sources, see Section A.1.4 in the Appendix) when we use only the distance to the nearest source. Also, if the sources are in the same phase, the early radial excitation (before any collisions) will also be synchronous, multiplying the size of synchronized neurons by the number of sources. Also the closer spatial distance gives less opportunity for variation due to noise disturbances. Knowing this we can try to minimize the spike distance, or to put it another way, increase synchrony by introducing more and more waves.

3.3 Optimal parameters

As mentioned in the last two sections, we can decrease spike distance by a) adding coupling to an uncoupled network with CR optimal noise, and b) using more noise in a strongly coupled network to increase the number of waves. So we might conclude that there is an optimal working regime of a certain noise level and a certain synaptic coupling.

A parameter search confirms this with best synchrony with a relationship of $\sigma \approx 20 \cdot w_{syn}$ and a minimal value of about $\sigma = 0.3$ and $w_{syn} = 0.02$

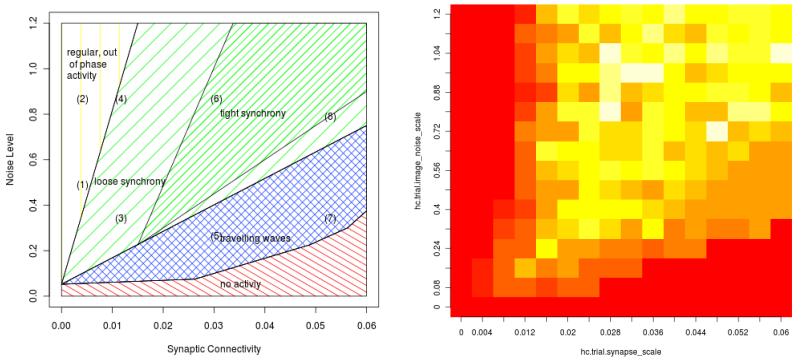


Figure 3.3.1: Parameter search over noise strength and synaptic coupling. On the left we have a qualitative assessment of activity, on the right the R_{syn} values for each parameter pair.

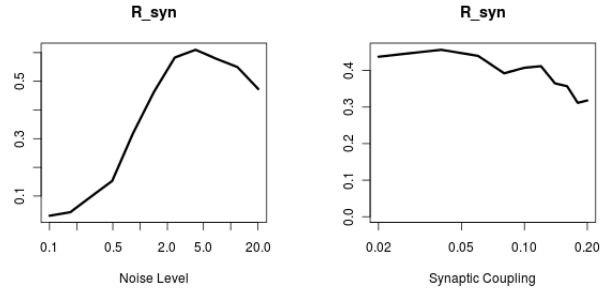


Figure 3.3.2: Taking the mean R_{syn} over one of the parameters of 3.3.1, we get a clear maximum for noise while the mean over all noise levels is less informative and the value for 0 is omitted due to logarithmic plotting.

3.4 Contour Segregation

3.4.1 Two arbitrary populations

If we now define some area to receive more noise then some other area, we should be able to read out the difference in the R_{syn} values and spike distances. In some cases this happens instantly, in others a change over multiple waves leads to different behaviour as for example in Figures A.1.5, A.1.6. In that simulation the background receives a noise level that is lower than the stimulus, but still high enough to lead to spontaneous spikes. Only the firing rate difference is responsible for gradually separating the two populations. The faster frequency leads to earlier spikes in the next wave for the neuron that spiked and those that are directly connected to it. More distant neurons are driven into a non-spontaneous mode as their reexcitation period is longer than the delay between waves.

Neurons in the visual cortex do respond with different rate codes to certain stimuli. Without this behaviour knowledge about the visual cortex and the features it responds to would be very thin. In our simulation on the other hand, the firing rate is almost equal for all neurons, as neurons are activated by waves if the stimulus is not strong enough to excite them in time. If we run the same simulation with the same noise for two different coupling strengths - one suitable for waves, one not - we get one set of activity that encodes the stimulus as a rate code and one that encodes the stimulus as a synchronization population as in Figure A.1.4 on page 34.

3.4.2 Contours

Using contours with different geometrical properties and optimal noise for the stimulus area, we can extract the stimulus to a sufficient degree that linear classifiers could distinguish them easily.

Figure 3.4.1: Examples for the different contours used.

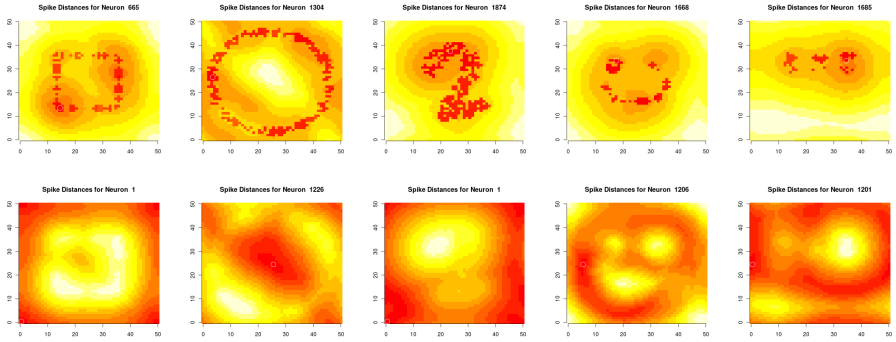


Figure 3.4.2: Examples for recovered stimuli. Red marks very low spike distance, as reference in the upper row stimulus neurons are used and in the lower row out-of stimulus neurons.

In the next sections we will examine why the synchronization works the way it does.

3.5 Simplified Explanatory Model

Comparing for different stimuli by how many sources and waves are needed to have a significantly lower distance to the stimulus than to non-stimulus neurons can be done in the simplified model defined in 2.4. There we neglected noise effects on wave transmission and assumed a uniform speed at which the waves propagate.

In the comparing plot 3.5.2, we can examine how many sources a certain stimulus needs for relative synchrony. A bold digit is for example very effective, while two distant lines or a hollow square need about 40% of neurons as synchronous sources to be clearly separable from the background.

A clear distinction can be seen if we take a random sample of neurons and compute the distance ratio, which will be above one until a very large portion of neurons are sources. This is because the radial propagation synchronizes the neurons surrounding the sources, which in a random selection of neurons are unlikely to be sources themselves.

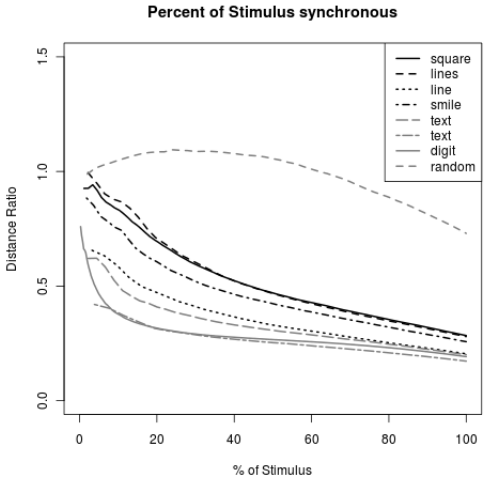


Figure 3.5.1: Using the simplified model and in each run increasing the number of sources: random stimuli do not bind, other stimuli bind, the more sources, the better. If we use half of the stimulus as a simultaneous onset, we have good separability for all but random stimulus cases.

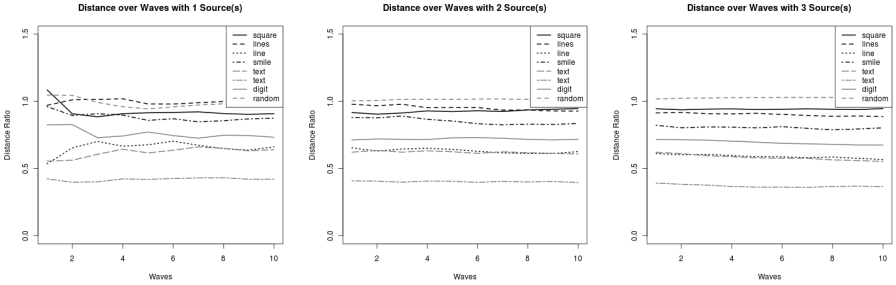


Figure 3.5.2: The changes from wave to wave are very small. If there is only one source, the subsequent waves provide additional information, most of it in the jump from first to second wave. Other than this the binding is not significantly enhanced with subsequent waves in the simplified model.

3.6 Dissonant Input

Taking the intuition from the simplified model that initial zero phase difference, ie. signal onset, has a strong influence on synchrony, we examined the difference between close and removed neurons and whether the synaptic connections can actually compensate misleading information. If the input is temporally

dissonant we would expect on large distances to establish two synchrony populations with different phase and for small distances for the phases to assimilate. Running the same simulation, but one time lesioning the lattice into two disconnected areas, using an asynchronous stimulus onset for both runs, we receive the distance maps in Figure, showing synchrony and asynchrony respectively.

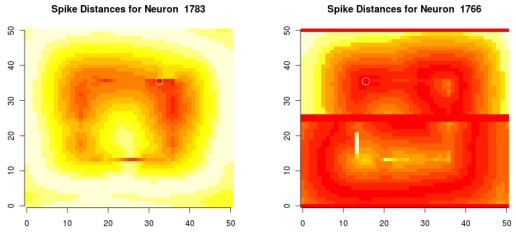


Figure 3.6.1: With a very large discrepancy (500 timesteps) the lesioned network developed divergent activity: the stimulus on the other side is more distant than all other neurons on that side, there has been no binding between the populations.

Chapter 4

Discussion

4.1 Mechanisms for Synchrony

Synchronization is achieved along strong connections with few interneurons, ie. similar features. The pattern that is to be segregated has to have a high degree of compactness - that is a lot of internal-internal connections and only a few internal-external connections eg. a closed, continuous contour.

Through the random choice of activity sources, the spike distances even over larger distances are very synchronous, suggesting that the visual system benefits from a synchronized signal onset, which might be achieved as a side effect of eg. saccades or a global inhibitory signal. This might be supported by the use of temporal cues to influence figure binding [Leonards et al., 1996]. Close regions can compensate for this dissonant input over time in our simulation.

In Section 3.1 and 3.2 we explained the synchrony as being the result of coherent frequencies of the oscillators and wave dynamics. Both progressions - adding coupling to same frequency oscillators and adding waves to strong coupled networks - lead to more synchronization.

The simplified model, that assumed n synchronous sources, showed how waves can synchronize an area around sources, but did not show improvement effects if, for more than one source, subsequent waves were computed (see Figure 3.5.2). This suggests that improvement in the actual simulation is mostly due to the spontaneous sources becoming more synchronous. In the case of asynchronous onset the sources in the top and bottom half start in very different phases, but gradually compensate for this.

4.2 Conclusion

Synchrony of coupled oscillators depends on the input to frequency mapping of each individual oscillator, including how stable the frequency is, and how strong coupling assimilates phase differences. In our case a gradual progression towards synchrony can be achieved as each wave brings the surround of the source into

a certain phase relation, which combined with the stochastic spiking properties of those neurons will equalize the phases of neurons with equal frequency.

Not only is the network capable of synchronizing itself within the stochastic boundaries, but it is even possible to synchronize certain regions with each other, while others remain unsynchronised. Those regions have to be connected within to benefit from Array Enhanced Coherence Resonance and to equalize their phases, which provides mechanisms to only bind *sensible* stimuli, that are actually composed of matching features (spatial in our case), and dismisses actual random stimuli (compare Figures 3.5.1 and A.2.1).

The question stands where in the brain could a noise coded signal as used in these simulations come from? If we consider the white noise to be the result of presynaptic activity of a very large population of neurons, an increase in rapid voltage gradients could be caused by more synchronous behaviour, ie. if we assume an uncorrelated distribution of N spikes and then move k spikes t coincide with some of the other spikes, we will no longer have pure white noise, but an increase in high frequencies which results in more excitable stimuli for our noise excitable oscillator. With this intuition we can conjecture that given a large population of neurons, a noise excitable neuron can detect synchrony in similar ways the R_{syn} measure or the Tempotron do (see Appendix Figure A.3.1).

4.3 Prospect

For future projects a number of topics might be interesting to pursue:

- What influence has the dimensionality on the restrictions of the system? The visual cortex is not at all connected to reflect only spatial relationships, but a number of visually important low level features. Reflecting those features in connectivity and choosing real world stimuli might give rise to dynamics that might bind objects based on those features.

- Related to that one could instead of using only local connections use weaker and/or delayed synapses to connect distant oscillators to influence the kind of temporal code produced.

- It is possible that a weaker mechanism than the proposed one throughout the visual pathway incrementally synchronizes the stimulus using horizontal connections to modulate transmission speed. Instead of multiple waves, multiple layers would equalize the phases.

- If we assume subsequent processing areas, those neurons could - via STDP - produce higher order features, which should be analysed in contrast with connections learned via rate codes in terms of what is learned and how fast.

- The investigation of arbitrary temporal codes in contrast to synchrony is another interesting aspect. Central Pattern Generators in motor sys-

tems can orchestrate complex movement patterns from simple input signals. In a genetic algorithm study to mimic salamander limb and body movement[Ijspeert, 2001], the simulated neural network converged to a travelling wave dynamic - although with more complex oscillatory circuits - in a 1d line of oscillators as in Figure 1.3.2. Without much engineering effort a travelling wave dynamic could be used to build a generator for any fixed length sequence of non-interactive outputs by converting each spike of an oscillator with an intermediate building block into the intended signal for this timestep, yielding a temporal progression of each output. In computer vision the properties of the mean field give possibilities to easily generalize over eg. geometric mirror or rotation operations. In one paper a similar network is used which discourages waves by a strong global inhibition as soon as the first spikes occur.

- The possibilities of passing on synchrony, as proposed in Figure A.3.1, should be explored in more detail.
- And it remains to show how the Tempotron and other analysis methods can be used on synchrony and arbitrary temporal codes to recover stimuli with and without prior knowledge.

Appendix A

Additional Material

A.1 Additional plots

A.1.1 Wave Activity

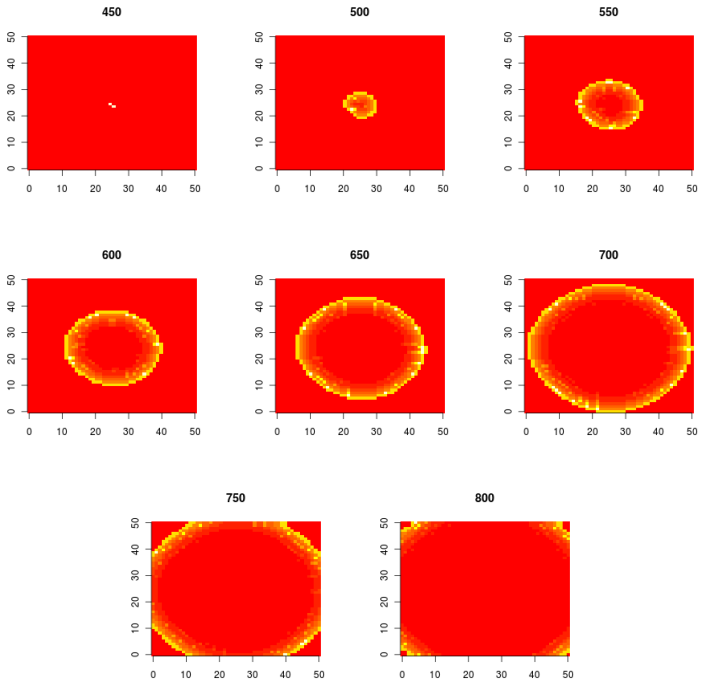


Figure A.1.1: Travelling Wave behaviour originating from the strongly stimulated center: image_noise_scale=0.2, bg_noise_scale=0, synapse_scale=0.015, stimulus: 3×3 square

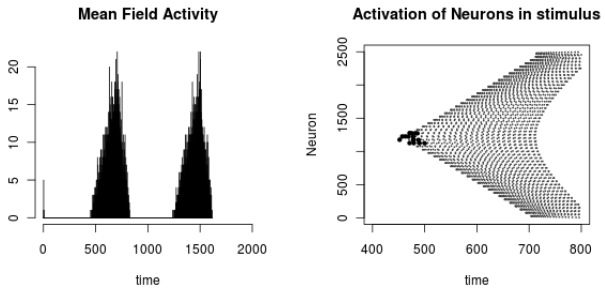


Figure A.1.2: Mean field potential of the activity of Figure 2.1.1 and A.1.1 and a raster plot of activity.

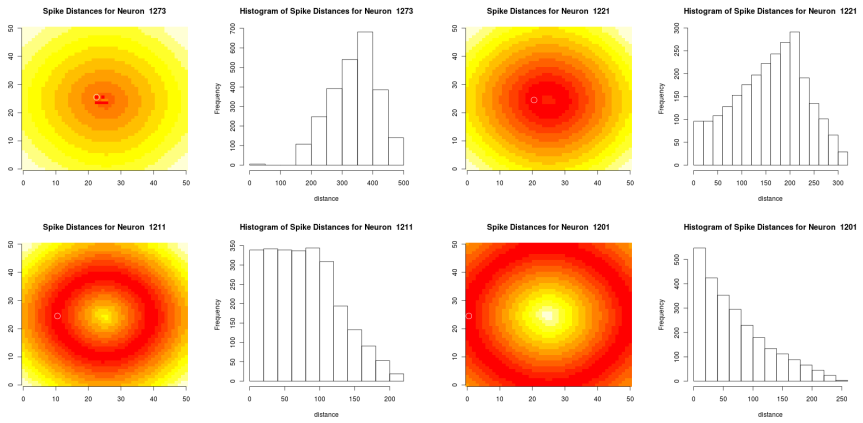


Figure A.1.3: Spike Distances seen from shown Neuron in the activity of Figure 2.1.1 and A.1.1. Dark red signifies low distance, bright/yellow high distance, so the dark rings around are populations with the nearly the same phase as the circled neurons across the whole time span. The neurons synchronous with 1273 are the sources of activity, all others are evoked avalanches.

A.1.2 Rate Coding in CR

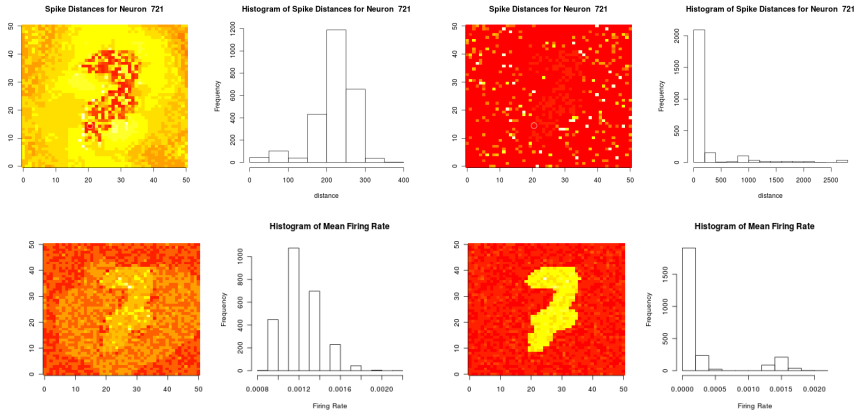


Figure A.1.4: Firing rate and synchrony population for a coupled (left) and uncoupled (right) network with identical noise. Note that the histogram of the coupled firing rate has no values below 8 spikes per 1000 time steps per neuron, while a lot of the uncoupled neurons have a firing rate close to 0.

A.1.3 Gradual Change

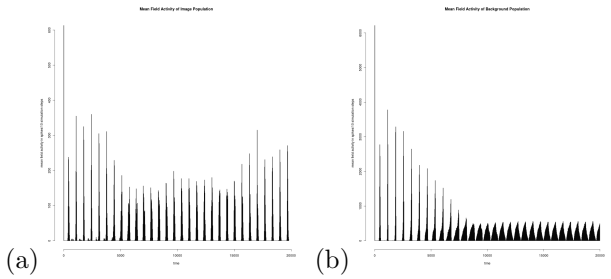


Figure A.1.5: Mean Field Activity of (a) the image population and (b) the background, clearly separating both populations

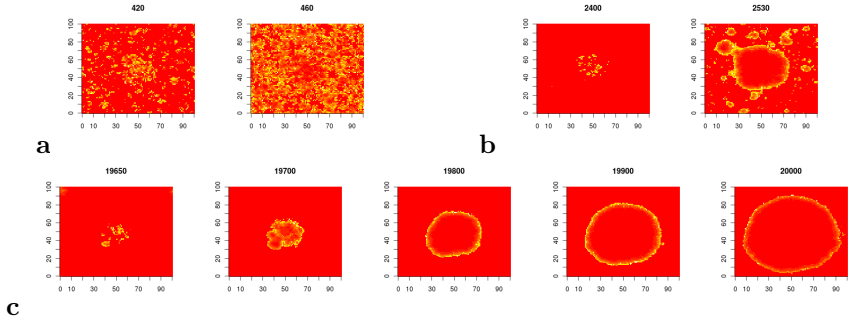


Figure A.1.6: Spreading activation in the first few cycles (a and b) and a later cycle (c) of Figure A.1.5

A.1.4 Big O Notation for Distance Scaling

To estimate how the distribution of distances and especially the mean distance change with additional sources, the following equation A.3 is assumed without proof:

$$d_1(a,b) = |a-b| \tag{A.1}$$

$$d_n(a,b_1,\dots b_n) = min|a-b_n| \tag{A.2}$$

$$\overline{d_n}(a,*) \overset{?}{\approx} \frac{\overline{d_1}(a,*)}{\sqrt{n}} \tag{A.3}$$

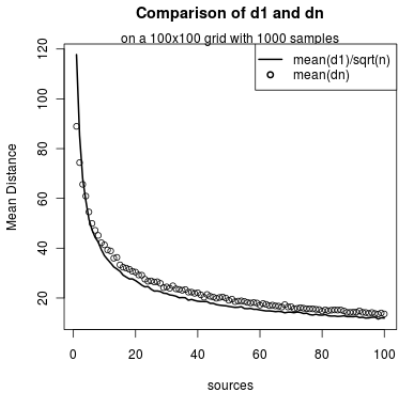


Figure A.1.7: A plot to compare $\overline{d_n}(a,*) \approx \frac{\overline{d_1}(a,*)}{\sqrt{n}}$ in a discrete 100x100 grid, using 1000 samples (10%) for the mean.

A.2 Random Stimulus

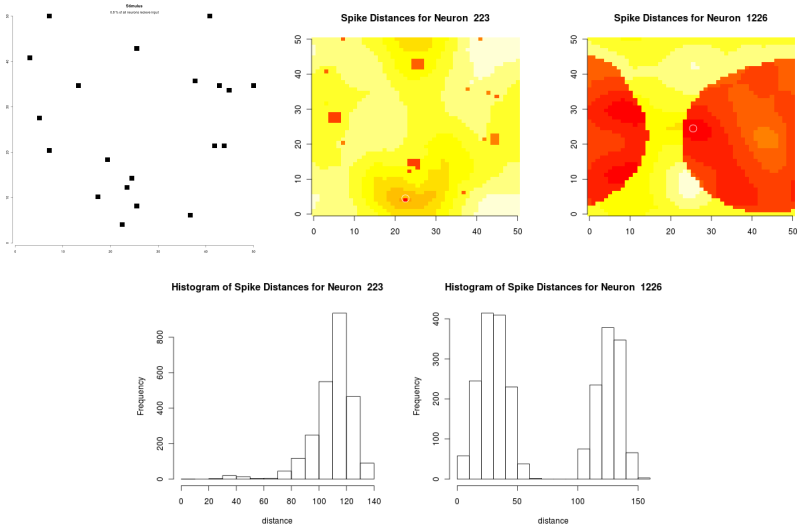


Figure A.2.1: A random stimulus with distance heatmaps for a stimulus and a non-stimulus neurons plotted. Note how the one-neuron wide stimulus regions are either neglected or overestimated even in the quite good map in the middle. The non-stimulus neuron shows a clear binding of two very arbitrary populations. Without actual knowledge about the stimulus it is more plausible to assume that the stimulus was a circle instead of the random drawn neurons.

A.3 Passing on Noise

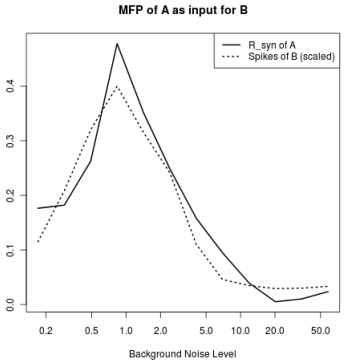


Figure A.3.1: The Mean Field Potential output of twelve simulations with increasing noise in a network A was used as input into an uncoupled network B. A small noise term was added, so that the activity could diverge and the spikes were counted. In this plot we see the R_{syn} value of each of the 12 simulations in A compared to the spike count of B that received this activity as input. This proves that the synchrony acts as a noise source, so with a careful tuning of parameters, network B could be capable of using this signal to pass on and/or enhance synchrony bindings.

A.4 R_syn

From the activity of any population we can compute a measure how synchronous it is called R_syn. Essentially it is the standard deviation of the mean field scaled by the mean standard deviation of each individual signal.

$$R_{syn}(x) = \frac{var_t(mean_n(x))}{mean_n(var_t(x))} \tag{A.4}$$

We first calculate the mean activity of all neurons over each timestep and see how much these values deviate from their mean. We can visualize this as the area between the mean field curve and the mean of that curve itself, however large deviations are weighted stronger since we square the difference.

The term is then normalized with the mean variance, that is the mean of each spike trains individual variance, which is roughly proportional to the number of spikes per time step.

If all signals are the same, the variance of the mean field is exactly the variance of each individual signal, giving us an R_{syn} of exactly 1. But if we have completely uncorrelated inputs, so that each signal has a high variance, but the mean field is nearly equalized and has therefore no variance, we get a value of nearly 0.

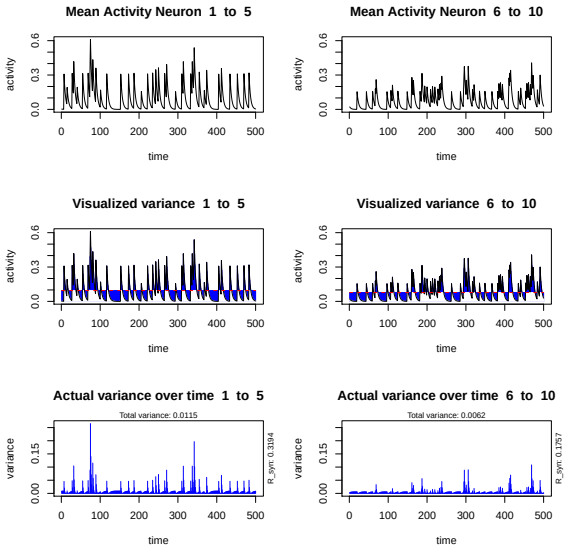


Figure A.4.1: A toy example of how R_syn is computed as produced by test_plots.r.

In the toy example in Figure A.4.1, the first population contains two spike trains twice, so it should be more synchronous than the other one. The outcome is a value of 0.32 and 0.17 respectively, so R_{syn} classified them correctly.

A.5 Binary R_syn

When we only work with spike-onset like events, we can simplify the R_syn term by restricting *vtt* to 0 or 1 for no activity or spike onset and simplify the mean variance in the denominator.

$$R\_syn\_fast = \frac{var(colMeans(vtt))}{mean(0.25 - (rowMeans(vtt) - 0.50)^2)}$$ [1]

This has the advantage, that the variance (the more costly operation) does not need to be computed for every single spike train. This method is only useful for a large number of signals, as the mean field will not vary very much if no spikes coincide.

[1] *var* is the variance of the whole vector, *colMeans* and *rowMeans* denote taking the mean for every column or row in the matrix producing a vector. *vtt* is organized with each row corresponding to one neuron and each column to one timestep. So *colMeans(vtt)* is a vector with length Max_{time} and *rowMeans(vtt)* with one value for each neuron.

A.6 Tempotron

The Tempotron is a method to learn temporal patterns. It works as a leaky integrate-and-fire neuron that detects previously learned temporal patterns by weighting input signals and measuring how often the integrate-and-fire neuron fires [Gütig and Sompolinsky, 2006].

Contrary to the R_syn measure, the Tempotron does not compute the variance, but since we add up kerneld and weighted synaptic input, we can see the similarity between the two methods. While the variance integrates area that deviates from the mean value, the tempotron integrates only those values that are higher than a certain value with a constant (eg. 1). With an appropriate kernel and matching threshold we can find synchronous activity in the same way we can use the variance. Note in Figure A.6.1 that different thresholds react differently to how close the spike trains are synchronous and might punish 0-phase synchrony in favour of near-0-lag synchrony.

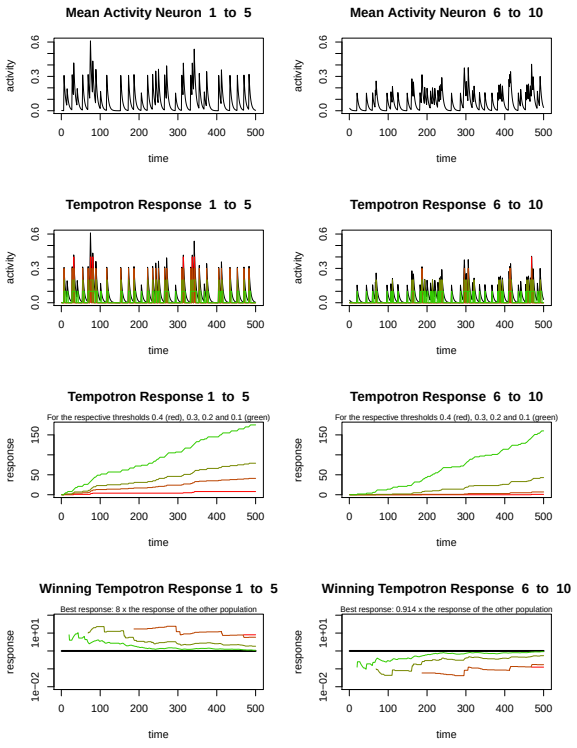


Figure A.6.1: A toy example of how the Tempotron response is computed as produced by test_plots.r. Different colors denote different thresholds. The last plot compares the tempotron response between the two populations for different thresholds.

But then again the Tempotron is more versatile if we converge from the idea of synchronous activity. For more complex patterns, eg. the detection of wave origins, appropriate parameters can be learned with appropriate labelled learning data. For simple radial wave detection, the connective matrix can also be created manually, with a simple on-center, strong off-surround dynamic which will cause the Tempotron to notice waves coming from the center, but ignore all travelling activity that originated from outside the local region. A more detailed account will be given by Clemens Korndörfer [Korndörfer, 2011].

Appendix B

Source Code

The source code can be obtained from http://huth-web.de/happy_cortex/hc_v10.zip, by email reqest to jahuth@uos.de or from the project dropbox.

```
hc_v10/
      README                            -- a readme file
      run.r                             -- the main script for simulation
      templates.r                       -- a configuration template file containing multiple simulation setups
      last_trial_number.r               -- contains the number of the last simulation
      test_plots.r                      -- simulates a toy example for R_syn and Tempotron
      plots/                            -- will contain the generated plots and data
            ...
      last_rd_number.r                  -- contains the last number of rd_simulations
      rd_plots/                         -- will contain the results of the simplified model simulations
            ...
      fhn/                              -- contains simulations to show FHN behaviour
            fhn_demo.r
            fhn_pulses.r
            coupling_diagram.r
            fhn_noises.r
      hc/                               -- contains internal code for loading stimuli and analysing spikes
            ...
      simplified_model/                 -- contains code to simulate the simplified model
            distance_distribution.r
            radial_distance.r
            analyse_distances.r
            tmp_distances.r
            radial_distance_batch.r
```

It is executable in R and should work without any installation, apart from the *animation* R package, that is required if animations are to be produced. It can be automatically installed with the R command:

```
> install.packages("animation");
```

Bibliography

[Engel et al., 1991] Engel, A. K., König, P., and Singer, W. (1991). Direct physiological evidence for scene segmentation by temporal coding. *Proceedings of the National Academy of Sciences of the United States of America*, 88(20):9136–40.

[Frank and Asuncion, 2010] Frank, A. and Asuncion, A. (2010). {UCI} Machine Learning Repository.

[Giacomelli et al., 2000] Giacomelli, G., Giudici, M., Balle, S., and Tredicce, J. (2000). Experimental evidence of coherence resonance in an optical system. *Physical review letters*, 84(15):3298–301.

[Gray et al., 1989] Gray, C., König, P., Engel, A. K., and Singer, W. (1989). Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature*, 338(6213):334–337.

[Gütig and Sompolinsky, 2006] Gütig, R. and Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing-based decisions. *Nature neuroscience*, 9(3):420–8.

[Ijspeert, 2001] Ijspeert, a. J. (2001). A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander. *Biological cybernetics*, 84(5):331–48.

[Izhikevich, 2007] Izhikevich, E. (2007). *Dynamical systems in neuroscience: The geometry of excitability and bursting*. The MIT press.

[Izhikevich and FitzHugh, 2006] Izhikevich, E. and FitzHugh, R. (2006). FitzHugh-Nagumo model. *Scholarpedia*, 1(9):1349.

[König et al., 1995] König, P., Engel, A. K., and Singer, W. (1995). Relation between oscillatory activity and long-range synchronization in cat visual cortex. *Proceedings of the National Academy of Sciences of the United States of America*, 92(1):290–4.

[Korndörfer, 2011] Korndörfer, C. (2011). *t.b.a.* Bachelors thesis.

[Lamme and Spekreijse, 1997] Lamme, V. and Spekreijse, H. (1997). Neuronal synchrony does not represent texture segregation. *Eos*, 78(November):229–233.

[Leonards et al., 1996] Leonards, U., Singer, W., and Fahle, M. (1996). The influence of temporal phase differences on texture segmentation. *Vision research*, 36(17):2689–97.

[Perc, 2005] Perc, M. (2005). Spatial coherence resonance in excitable media. *Physical Review E*, 72(1):1–6.

[R Development Core Team, 2011] R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*.

[Singer, 1993] Singer, W. (1993). Synchronization of cortical activity and its putative role in information processing and learning. *Annual Review of Physiology*, 55(1):349–374.

[Singer and Gray, 1995] Singer, W. and Gray, C. M. (1995). Visual feature integration and the temporal correlation hypothesis. *Annual review of neuroscience*, 18(1):555–86.

[Ullner et al., 2008] Ullner, E., Vicente, R., Pipa, G., and Garc\'\ia-Ojalvo, J. (2008). Contour integration and synchronization in neuronal networks of the visual cortex. *Artificial Neural Networks-ICANN 2008*, 5164/2008:703–712.

[Wang and Terman, 1997] Wang, D. and Terman, D. (1997). Image segmentation based on oscillatory correlation. *Neural computation*, 9(4):805–36.

List of Figures

Versicherung an Eides statt

Hiermit erkläre ich diese Bachelor Arbeit selbständig verfasst und alle verwendeten oder zitierten Quellen, Hilfsmittel, etc. angegeben zu haben.