UNIVERSITY OF ZURICH AND ETH ZURICH

Master Thesis

Memory for serial order in spiking neural networks

A songbird-inspired spiking dynamic neural field model for sensorimotor sequence learning

Author:

Alpha Renner

Supervisors:

Yulia Sandamirskaya

Giacomo Indiveri

A thesis submitted in fulfillment of the requirements

for the degree of MSc UZH ETH in Neural Systems and Computation

in the Neuromorphic Cognitive Systems Group

Institute of Neuroinformatics

January 2017

Declaration of Authorship

I, Alpha Renner, declare that this thesis titled, “Memory for serial order in

spiking neural networks” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Abstract

Sequences are ubiquitous in any kind of behaviour from communication (syllables, words, sentences) to movements. Zebra finches learn a song consisting of several syllables at young age from a tutor and replay it later in a stereotyped manner. I present a spiking neural network model for learning sound sequences in a closed sensorimotor loop, inspired by bird song learning. The main building block of the model are dynamic neural fields (DNFs) implemented as spike-based winner-take-all networks (WTA).

On the sensory side of the model, a feature-based DNF representation of the sound is mapped to a 2D perceptual space using a spiking self-organizing map. On the motor side, sound is represented in a 2D field and generated using a model of the bird's vocal organ (Syrinx).

A mapping between the perceptual and the motor space is learned using STDP synapses. Sequences of sounds are acquired on two hierarchical levels:

(1) Each syllable in a song is represented as a trajectory in the sensory space that is mapped to an ordered population of chain neurons that creates a representation of time. Using auditory feedback, neurons of the chain are then mapped to their corresponding motor representation;

(2) A sequence of syllables is learned in a DNF-based serial order model.

Overall, the model can learn to produce sounds and to reproduce the presented tutor sequences. I have tested my system on artificial stimuli so far and envision a neuromorphic closed-loop implementation.

Content

Introduction

Motivation for this project

I started this project, because I wanted to know more about dynamic neural fields (DNF), as they possess an intriguing power to propose potential mechanisms for simple and complex cognitive functions. I also realized that they, what is even more intriguing, allow a modular structure and therefore an intuitive understanding of the working principles for even large and complex architectures. Which is something that probably comes with costs, but is also something I miss in state-of-the-art machine learning approaches: although, in my previous short project, I found some interpretable cells (like direction cells) in a maze solving LSTM-based recurrent neural network, it remains, like our brains, an "opaque box".

As almost all cognitive functions involve time and therefore sequential activity, it seemed natural to me, to deal with sequence representations in the scope of dynamic field theory (DFT), especially as my supervisor recently extended the theory by introducing a serial order model capable of learning sequences and autonomously switching through a series of attractor states in a dynamic field architecture (Sandamirskaya and Schöner 2010).

As I am not only interested in the theoretical model of sequence learning, but also in applying it to a neurobiological problem, I chose to develop a simple model of vocal learning inspired by the commonly used model animal of vocal learning, the zebra finch. Inspired by previous models of birdsong learning (Doya and Sejnowski 1995, Leonardo and Fee 2005, Hanuschkin, Ganguli et al. 2013) (see Fig. 2) and a DNF-based model for learning of sensorimotor transformations (Rudolph, Storck et al. 2015). I designed a model combining sequence learning with the learning of a sensorimotor map which is necessary to produce a song.

I realized that the commonly used elements of dynamic field theory (commonly used for cognitive functions and not for lower level motor processes) are in fact not "dynamic" enough to tackle fast continuous trajectories through a dynamic field as required to produce a song syllable. Therefore, having the sequential bursting of the songbird nucleus HVC (Hahnloser, Kozhevnikov et al. 2002) and also internally generated cell assembly sequences in mammal hippocampus (Pastalkova, Itskov et al. 2008) in mind, I introduced a synfire chain into the model that is able to almost continuously drive a peak through a dynamic field allowing for sensorimotor trajectories. This synfire chain could be integrated into DFT as a one dimensional dynamic field with an asymmetric connectivity kernel which leads to a drift in one direction. It potentially provides the lowest level of the DFT-based sequence learning hierarchy, that has a flexible behavioural organisation (Richter, Sandamirskaya et al. 2012) on top and the more rigid but controllable serial order model (Sandamirskaya and Schöner 2010) on an intermediate level (see Fig. 1).

Although DFT provides the way of thinking about the model, for simulations, I use an implementation in spiking neural networks. From a theoretical point of view, this does not provide a clear advantage, on the contrary, it poses a challenge as spiking networks are not easy to handle and ready to use modules were lacking. Nevertheless, convinced by the idea that dynamic field theory can provide a

powerful framework for the implementation of cognitive functions into spiking neuromorphic hardware (Sandamirskaya 2013) and willing to advance and test this idea, I was ready to take the challenge.

In the following I will introduce dynamic field theory as dynamic fields are the basic building block of the model and the theory provides the framework of the model presented in this thesis. Subsequently, I will review findings in birdsong learning that have inspired the current form of the model.

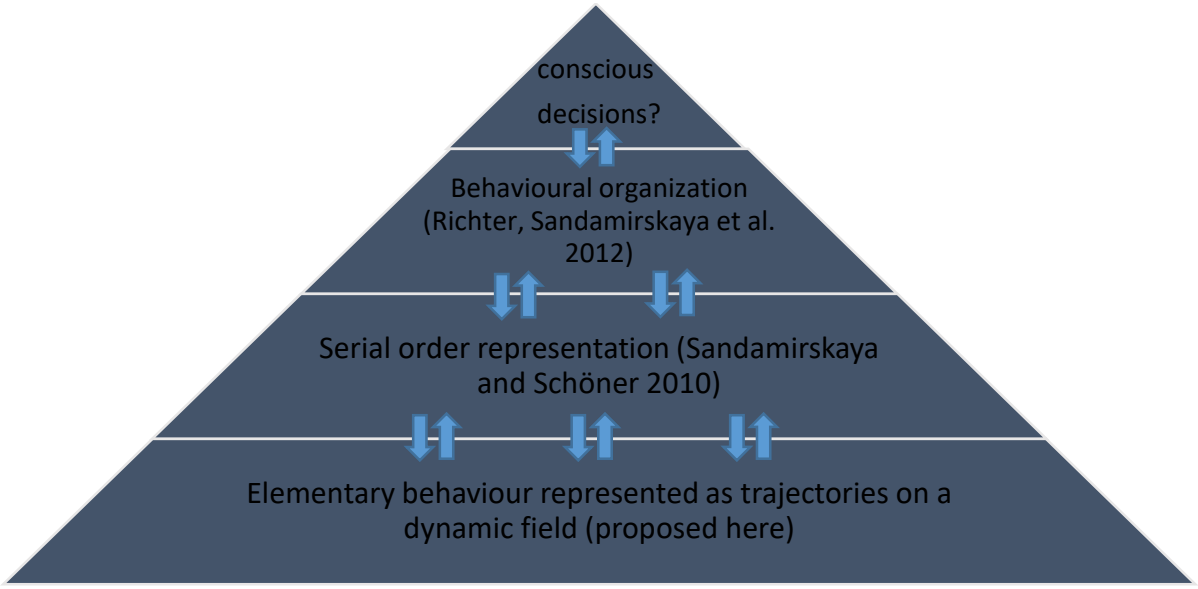

Fig. 1 Proposed hierarchy of DNF-based behavioural sequence representations. On the lowest level trajectories in sensorimotor fields are driven by sensory input and internally generated sequences and drive muscle output. On a higher level, a sequence of such elementary behaviours is represented in a serial order model. On an even higher level more complex behavioural sequences are represented in a behavioural organization that implements certain constraints but still remains more flexible and can be learned with reinforcement learning. The different hierarchies act on different time scales and have different levels of control. While the lowest level drives short elementary movement sequences in the second or sub-second range behavioural organization sets the schedule for behavioural sequences in the range of minutes or hours. On the highest level, I put "conscious decisions", as it seems to be the highest level of behavioural control, although, at the moment, it is unclear what that even means, therefore the question mark…

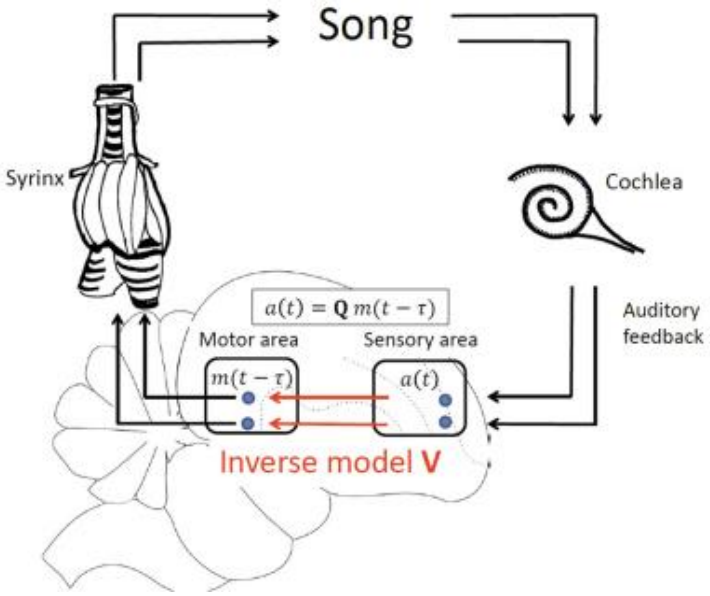

Fig. 2 Schematic of a simple model of birdsong learning in a sensorimotor loop. An inverse model of sound production that maps sensory states to their causing motor commands is learned using auditory feedback. Figure from Hanuschkin, Ganguli et al. (2013).

Dynamic neural fields

Dynamic Field Theory (DFT) is a powerful framework for modelling cognitive functions based on dynamical systems theory. Its main idea is that computation is performed in coupled low dimensional continuous activation fields in which activation moves towards attractors that are dynamically changing with input.

Apart from having universal computational power (Maass 2000), and substantial potential to propose plausible mechanisms for complex cognitive functions, the theory makes the idea of computation in dynamical systems more tractable as it uses the dynamic neural field (DNF) as its basic computational unit. This modular structure allows for an intuitive understanding of the ongoing computation and the construction of large and complex architectures for cognitive problems. This said, it could be viewed as the basis for a "neural programming language" especially for neuromorphic systems. As DNFs are computationally equivalent to a soft winner-take-all (WTA) network, which allows a biologically plausible implementation in spiking neural networks and also in neuromorphic hardware (Sandamirskaya 2013), the theory has the potential to unlock unused capabilities of neuromorphic spiking networks and might stimulate their scaling up.

A DNF is a continuous field of activation with excitatory interaction between close field locations and inhibitory interactions between remote locations. But let us start from the beginning: In the following short introduction to DFT, which is mainly based on the book from Schöner and Spencer (2015), I first show the dynamics of a single one dimensional self-excitatory unit. Based on this, dynamic fields are introduced and how they can implement elementary cognitive functions, such as working memory and attention. Finally, I will briefly present a DFT based model for sequence learning.

One dimensional dynamic system

A single self-excitatory activation variable already shows basic features of a dynamic field and is therefore ideal for an introduction. The following differential equation governs the activity of such a unit:

$$\tau \frac{du}{dt} = -u + h + s(t) + c * g(u) \quad (1)$$

with

$$g(u) = \frac{1}{1+\exp(-\beta u)} \quad (2)$$

u is the activation variable whose rate of change depends on the activation u itself, and an input that changes with time (s(t)). The first term -u leads to an exponential decay of the activation. The second term h is the resting level of the activation, which means the level to which the activation decays. Typically, h is negative. The fourth term, $c * g(u)$, is a self-excitatory term. Self-excitation is sigmoidal, which means that it does only take strong effect above a threshold, which is by convention 0.

When there is no or weak input, the system has a stable fixed point at the resting level h (h+s, when there is input) as it decays to this level. When there is intermediate input, the system shows bistability

as it has two stable and one instable fixed point. The instable fixed point marks the threshold at which the unit can be brought from resting level to the excited state. When the input gets even stronger, and h+s is above the self-excitation threshold the system becomes monostable again with a fixed point at a positive activation. This is explained graphically in Fig. 3.

The transformation from bistability to monostability is interpreted as a decision: The system decides in favour of one of the two fixed points. When a decision has happened the activation remains at the fixed point, even if the input changes slightly and the system goes back to bistability. Only when the it bifurcates again to the other monostable regime, the activation switches to the other fixed point, so the system shows a hysteresis, called decision hysteresis. The activation levels at which the switch happens are called detection instability as there, the detection of significant input takes places.

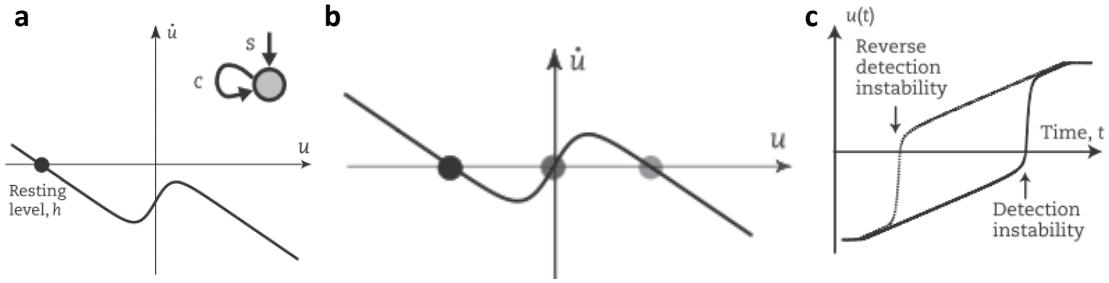

Fig. 3 Dynamics of a self-excitatory one dimensional system without (a) and with (b) external input. The form of the curve arises from the sum given in Eq. 1. Without input, the system has one fixed point at the resting level. With constant input, there are three fixed points and the system becomes bistable at the resting level and the excited level. The instable fixed point in between marks the threshold above which the system goes up to the excited state. When the input increases further the lower two fixed point disappear and only the excited state is stable. This leads to an effect called decision hysteresis (c): when the external input gets stronger, u̇ is shifted upwards and only when the resting state fixed point vanishes, the system moves to the excited state. Even when external input goes down again, it stays there until the input is so low, that the system undergoes bifurcation again and the excited fixed point vanishes. Figures from Schöner and Spencer (2015)

Dynamic neural fields

DNF work very similar to the single activation variable. A DNF is a continuous field (analogous to fields in mathematics and physics) of activation. Dynamics of this activation are specified by a differential equation that links the rate of activation change of every field location to the current activation level of the whole field and input to the field.

$$\tau \frac{du(x,t)}{dt} = -u(x,t) + h + s(x,t) + \int k(x-x')g\big(u(x',t)\big)\,dx' \qquad (3)$$

The first 3 terms of the equation are equivalent to the ones in the single activation case above, apart from dependence of the terms on the location in the field x. The fourth term defines interactions between the activations at the different locations in the field. The rate of change at every location x depends on the activation at every other location x' (therefore the integral) dependent on the distance of the two locations. Interaction is excitatory for short distances and inhibitory for long distances. Due to this, simulation of a field is possible with a Mexican hat shaped convolution kernel.

Analogous to the single activation case, the dynamic field shows mono or bistability dependent on the input (see Fig. 5). But the attractor is now not just a fixed point, but a curve with peaks at locations of input. These peaks of activation can be interpreted as objects. Depending on what a field stands for, the location of peaks codes for the value of a parameter or feature (like e.g. physical location). The broadness of a peak can specify precision of a perceptual state or belief.

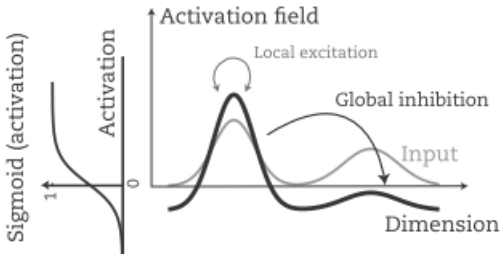

Fig. 4 In an activation field, there is local excitation and global inhibition. Peaks are stabilized from decay by excitation of near regions and from diffusion by inhibition of more remote regions around the peak. Smaller peaks and noise are suppressed by global inhibition. This behaviour is characteristic for a soft-winner-take-all computation. Figure from Schöner and Spencer (2015).

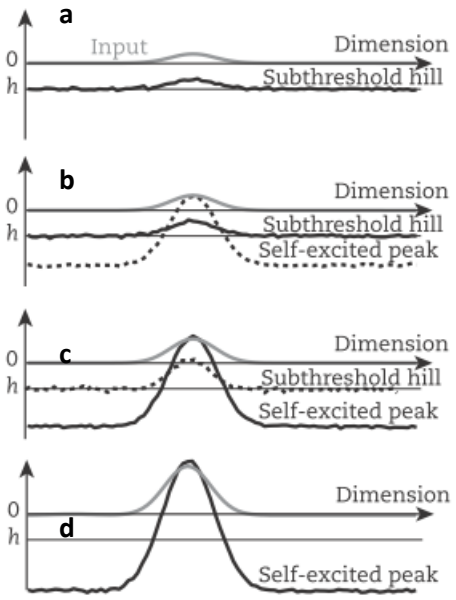

Fig. 5 Attractor states of a dynamic field. Inputs are grey and attractor states of field activation are black (dashed or solid). (a)a small localized input leads to a small hill that goes away, when input goes away as it exactly mirrors the input to the field (minus the resting level). (b) when the input becomes larger, self-excitation kicks in and a second attractor state, the self-excited peak, emerges. However, the system remains close to the bifurcation that lets the self-excited peak vanish again when input goes down. (c) With an even stronger input the self-excited peak becomes stable against input variations (d) For very strong input, the system is again monostable with only the self-excited peak as an attractor. These attractors can be seen in analogy to the ones presented in the one dimensional system in Fig. 3. Figure from Schöner and Spencer (2015).

Already with a single field, elementary cognitive functions like working memory can be modelled:

When self-excitation is so strong, that even without external input, there are 3 fixed points, the system is able to "store" it's activation once excited even when the input goes away. A reset is possible by inhibitory input (e.g. when a peak at another location is formed). This mechanism can be used to store a working memory of previous activations (see Fig. 6).

Another kind of short term memory in DNFs with normal self-excitation emerges due to the fact that when the input peak moves, the activation field peak naturally lags behind, so the trajectory is low pass filtered.

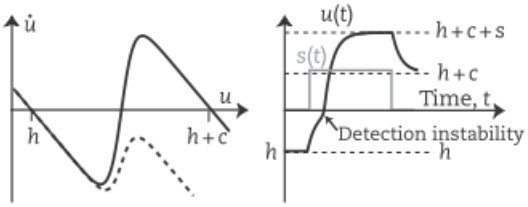

Fig. 6 Strong self-excitation leads to self-sustained peaks serving as working memory. Here, as a simplification, a single activation variable is shown. The self-excitation is so strong, that the system has three fixed points even when there is no input (compare Fig. 3a). So once in the excited state, the activation does not go back to the resting state even if the input vanishes. Figure from Schöner and Spencer (2015)

With two bidirectionally connected fields, with different self-excitation parameters, categorization can be modelled (see Fig. 7). The strongly self-excitatory memory field remembers past above-threshold peaks from the activation field as self-sustained peaks. Through the bidirectional connections, the activation field is constantly weakly activated by the peaks in the memory field. This sub-threshold activation is called preshaping, as it preshapes the activation field even without external input and makes it more easily excitable at those locations. If there is now a localized input in close proximity of one of the preshaped regions, the system responds with a peak at the preshaped region instead of the exact location of the input. This can be interpreted as categorization, as all inputs close to one of the preshaped locations are put in the respective category.

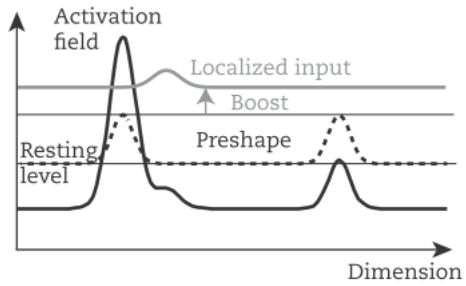

Fig. 7 Preshape and boost leading to a decision for a learned category. The activation is pre-activated (preshaped) by an input (e.g. by a memory field) at specific locations which makes it more excitable at those locations. A localized input close to one of the preshaped regions, elicits a decision for one of the preshaped peaks when the system is "asked" for a decision by a boost, which means a homogenous input that shifts the whole activation up, lifting subthreshold activations above threshold. Figure from Schöner and Spencer (2015)

Apart from constrains on the time scale of the dynamics, DNFs can be implemented as spiking soft winner-take-all (WTA) networks. Fields are no longer continuous, but an ordered population of neurons that has excitatory lateral synapses and global inhibitory synapses. Global inhibition is often modelled with an additional population of inhibitory neurons in order to satisfy the biological constraint that neurons are either excitatory or inhibitory but not both.

This implementation in spiking networks also provides the link from dynamic field theory to the brain. DNFs can be interpreted as populations of neurons organized in cortical feature maps that might have emerged through self-organization (see methods) (Schöner and Spencer 2015). Soft winner-take-all computations are not only considered as a crucial part of cortical processing (Riesenhuber and Poggio 1999, Douglas and Martin 2004), but are also part of virtually every state of the art deep and recurrent network used in machine learning (there called softmax). WTA networks are also used as core components of neuromorphic electronic circuits (Indiveri, Murer et al. 2001, Indiveri, Chicca et al. 2009).

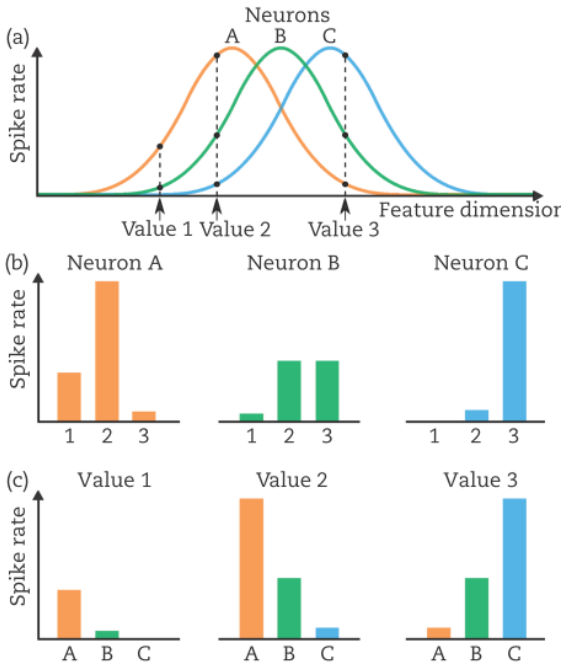

Fig. 8 Proposed neuronal implementation of dynamic neural field with rate-based population coding. (a) Each neuron has a Gaussian receptive field over the encoded feature dimension (tuning curve). (b) a single neuron does not uniquely code for a value in the feature dimension, as two values can elicit the same activity in the neuron. (c) If we look on all neurons however, the stimulus value is uniquely encoded (population coding). Looking at the whole population in an ordered way (as a field), there is a peak around the neuron that responds most strongly to the stimulus value. Figure from Schöner and Spencer (2015)

DFT can be used to construct complex architectures and solve problems as obstacle avoidance, reference frame transformations and space-feature and feature-feature binding (Schöner and Spencer 2015). It can also be used to model and learn sequences (Sandamirskaya and Schöner 2010), which is described in the next part.

Sequence learning with DNF

The DFT based sequence model presented by Sandamirskaya and Schöner (2010) represents a fixed sequence that can be linked to arbitrary content. The model consists mainly of two sets of interconnected nodes. Ordinal nodes represent an ordinal position in a sequence. Each ordinal node is associated with a memory node that stores, together with the other memory nodes the current state of the sequence (see Fig. 9). Ordinal nodes have self-excitatory and mutual inhibitory connections, so an activation of a node is stabilized and only one ordinal node can be active at the same time. When it is active, an ordinal node excites its own memory node which in turn inhibits its ordinal node a bit and excites the next ordinal node. Also memory nodes have self-excitatory connections and remain active once activated until the whole sequence is reset that is why they are called memory nodes.

The actual sequence learning happens, when the ordinal nodes are connected to specific locations of a content field using e.g. Hebbian learning. Locations in the content field can e.g. stand for elementary behaviours and link to the motor system. The motor system is assumed to be outside of the model and receives e.g. a goal state of the movement. In this thesis, an elementary behaviour is a trajectory in motor space (see methods section).

The switch of activity from one ordinal node to the next happens when the CoS-node (condition of satisfaction) inhibits the ordinal nodes so that the mutual inhibition vanishes and the last activated memory node can activate the next ordinal node. The CoS-node becomes active whenever an elementary behaviour that was elicited by the content field is finished and the system is ready to perform the next.

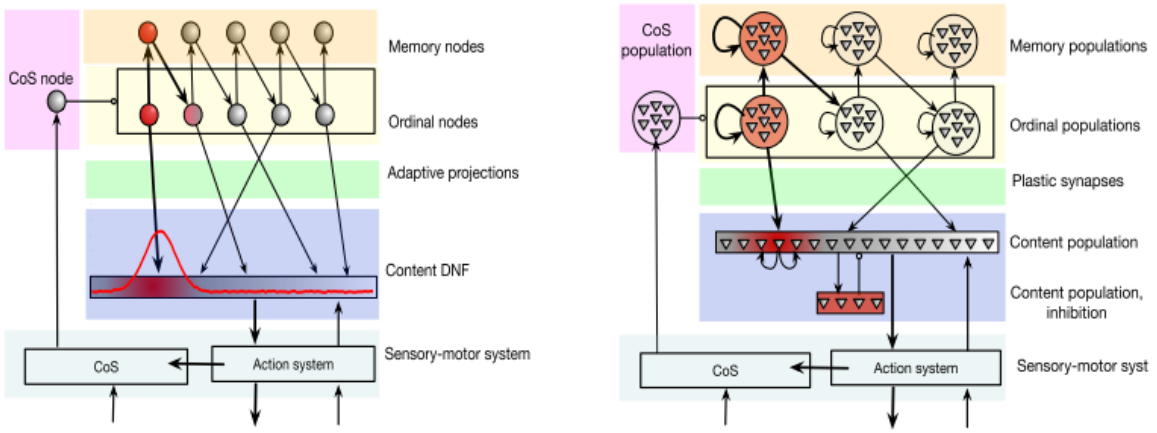

Fig. 9 DFT based serial order model. The left part shows the implementation of the model in DNFs, the right part with populations of neurons, like in a spiking network simulation or in neuromorphic hardware. Nodes are realized as small populations of neurons. This is especially important for neuromorphic implementations due to noise and mismatch. Ordinal nodes represent an ordinal position in a sequence. Each ordinal node is associated with a memory node. Both memory and ordinal nodes have self-excitatory connections. An active ordinal node excites and activates its memory node that inhibits it in turn and excites the next ordinal node, so that when the CoS node inhibits the ordinal population the next ordinal node in the row gets activated. Figure from Sandamirskaya (in preparation)

Song learning in zebra finches

Zebra finches (*Taeniopygia guttata*) are Australian birds that are studied intensively for decades in order to investigate vocal learning and sensorimotor integration as their male individuals have a very stereotyped short song that they learn from a tutor during a distinct learning period (critical period). Like in humans, there are different developmental phases of vocal learning and production (see Fig. 11): At first, the bird begins to babble (similar to human babies) i.e. utter highly variable sequences of tones. This so called subsong gradually develops, in the adult, to a crystallized song that is a stereotyped sequence of distinct syllables very similar to the tutor song (Doupe and Kuhl 1999).

The most prominent song related brain regions of the zebra finch are HVC (formerly known as high vocal centre, now used as a proper name), RA (robust nucleus of the archistriatum) and the anterior forebrain pathway (AFP) consisting of striatal Area X, DLM (medial nucleus of the dorsolateral thalamus) and the cortical structure LMAN (lateral magnocellular nucleus of the anterior nidoapallium) (see Fig. 10).

HVC receives auditory input and projects to RA and Area X. Neurons in HVC fire both during vocal activity and during listening to a song (Prather, Nowicki et al. 2009). RA drives the motor areas in the brainstem. By lesion experiments it has been established, that HVC and RA are necessary for song production. The AFP is seen as analogue to the cortico-basal ganglia pathway in mammals (Farries and Perkel 2002). Lesion studies show that it is necessary for song learning and song plasticity, but not for production of the crystallized song (Scharff and Nottebohm 1991). When auditory feedback is removed by deafening of an adult bird, vocal production slowly deteriorates, however, when LMAN is lesioned, this effect is prevented (Brainard and Doupe 2000). It was therefore hypothesized that LMAN produces exploratory variability that allows learning in juveniles and adaptive changes of the song in adults using auditory feedback.

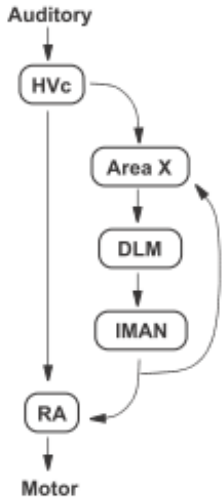

Fig. 10 Schematic of the zebra finch song system. HVC receives auditory input and neurons from HVC project to RA and Area X. RA drives muscle controlling motor regions of the brain. Area X, DLM and LMAN constitute the anterior forebrain pathway that is necessary for variability in the song. Figure from Dave, Yu et al. (1998)

Recent studies suggest, that there is a transition from LMAN driven variable subsong to HVC driven crystallized song (Aronov, Andalman et al. 2008) during development of young birds that might happen through a relative spike timing dependent competition between HVC and LMAN (Mehaffey and Doupe 2015). LMAN does not just randomly drive RA but can bias song production towards a certain rewarded goal (Andalman and Fee 2009) or away from a punished pitch (Tumer and Brainard 2007).

There is evidence for both hierarchical and non-hierarchical song structure (Glaze and Troyer 2006). Non-hierarchical models assume that the song is driven by clock like bursting in HVC.

A song consists of a couple of different motifs. Each motif (about 1 s long) consists of syllables and gaps. A syllable is a short (around 50-200 ms) tone or combination of tones and usually corresponds to an air sack pressure pulse (Franz and Goller 2002).

It was thought, that song hierarchy is also represented in the bird's brain: HVC might be responsible for the syllable sequence and RA for the representation of individual syllables, as electrophysiological disturbance of RA causes disturbance of syllables, but not of the song structure. Unlike disturbance of HVC that causes instabilities in both syllables and song structure (Vu, Mazurek et al. 1994).

However, this hierarchical view was challenged by Hahnloser, Kozhevnikov et al. (2002) by showing that HVC neurons projecting to RA burst sparsely and reliably at a single time in a motif, presumably driving more complex sequences of firing patterns in RA (see Fig. 13) and representing a "state" in the sequence instead of inducing a whole syllable. They suggest that HVC_{RA} Neurons fire sequentially and form an explicit representation of time. Existence of such a chain in HVC has been shown by Long, Jin et al. (2010). Internally generated neuron sequences have also been found e.g. in (rodents') hippocampus (Pastalkova, Itskov et al. 2008) and (primates') premotor cortex (Crowe, Zarco et al. 2014). Markowitz, Liberti III et al. (2015) showed that this temporal sequence of HVC projection neurons is also represented in the spatial organization of HVC neurons by temporal clusters.

RA generates a highly stereotyped and precise pattern of about 10 ms bursts which drives vocal (syringeal) motor neurons and respiratory areas (Wild 1993). As they are hard to record, there is basically not data about the dynamics of those motor neurons.

In this model, presented by Leonardo and Fee (2005), that has been called "music box model" synapses from HVC to RA supposedly contain the "melody" of the song in an abstract motor space and synapses from RA to the motor areas contain the mapping from this space to the actuators. There is no structure that determines the beginning of a sequence or discriminates syllables and gaps. This model is further supported by cooling experiments. When HVC (functioning as the clock) is cooled down and thereby slowed down, song speed is slowed down accordingly while maintaining the acoustic structure, cooling down RA has no such effect (Long and Fee 2008).

However, Glaze and Troyer (2006), using the fact that song length varies systematically over the course of the day, made measurements of syllable timing and found that song length changes are

not due to accumulation of independent shifts during the song but are present across the whole song. Furthermore, there is evidence that gaps and syllables are represented distinctly in the song system, as gaps are able to change their length stronger than syllables. Syllable onset seems to be triggered/synchronized externally as syllable length has a stronger trade-off with the length of the subsequent than with the preceding gap (Glaze and Troyer 2006).

This indicates, that gaps might just be the left over from the end of the syllable to the trigger of the next one and have no separate representation.

To reconcile these two views Glaze and Troyer (2006) suggest that there might be input (from thalamus or midbrain) to HVC that drives temporal grouping. The suggestion is supported by results of Schmidt (2003) who found synchronization of the two not interconnected HVC hemispheres to be especially high at the onset of syllables. However, probably this input region would need to have an independent clock. Alternatively, interneurons of HVC/RA or also LMAN, in analogy to the role of basal ganglia in sequence generation (Aldridge, Berridge et al. 2004), might be involved in structuring the sequence and lead to the temporal grouping. Also a recurrent activation from respiratory brainstem areas can provide timing information to the song control nuclei (Ashmore, Wild et al. 2005).

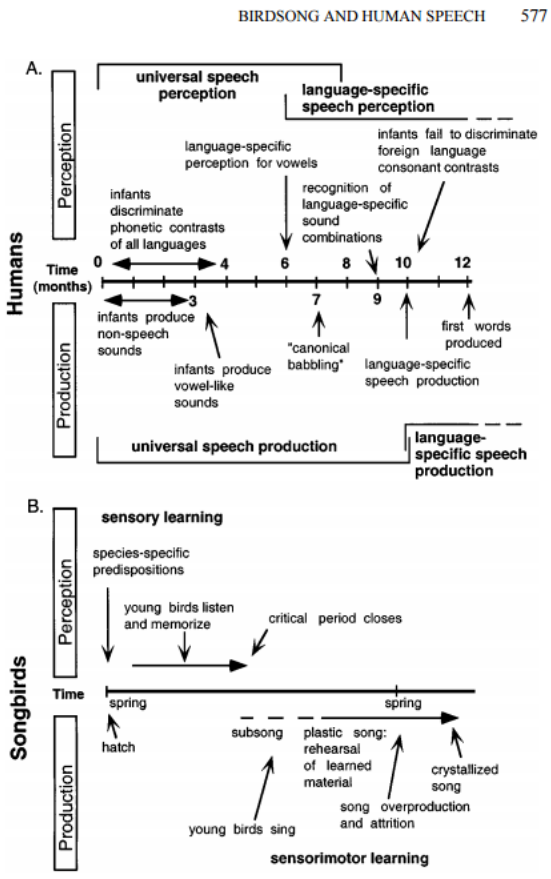

Fig. 11 Vocal sensorimotor learning in humans and songbirds. In both species, learning happens in distinct developmental phases. Figure from Doupe and Kuhl (1999).

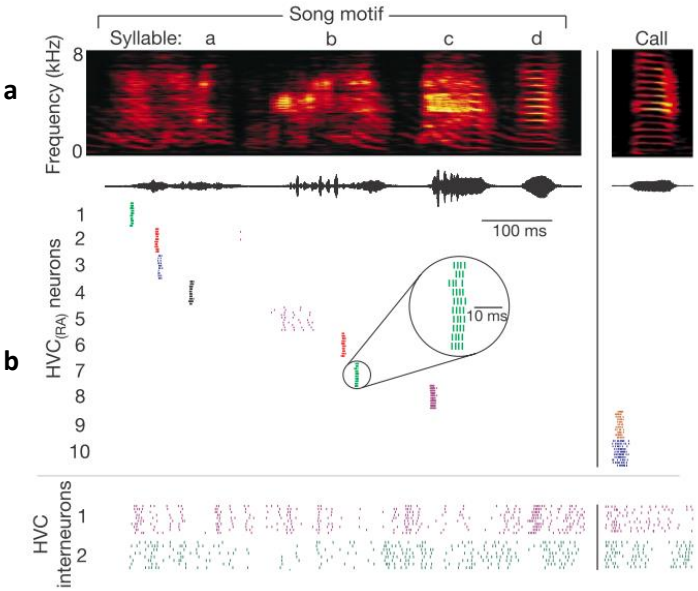

Fig. 12 HVC activity during singing. (a) spectrogram of a repeated song motif during which activity of HVC neurons was extracellularly recorded. 4 Syllables that constitute the motif are shown. (b) Raster plot of 10 HVC_{RA} neurons recorded for about 10 renditions of the motif. Although only a small subset of all HVC_{RA} neurons could be recorded, it looks like a sequence of about 10 ms bursts. Each neuron generates a single burst during each rendition of the motif. (c) Densely firing HVC interneurons. Figure from Hahnloser, Kozhevnikov et al. (2002)

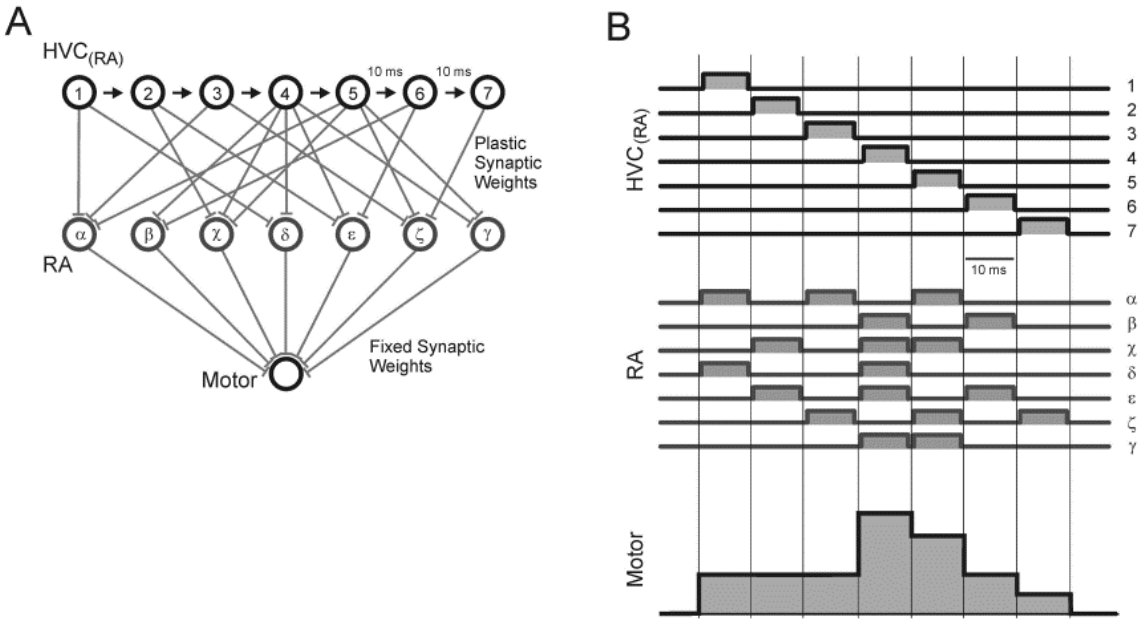

Fig. 13 Visualization of the "music box model". HVC produces a sparse temporal sequence of bursts that drives denser firing in RA that in turn activates motor regions. The melody and rhythm is stored in the plastic synapses from HVC to RA. End to end alignment of onsets and offsets is only for graphical clarity. Figure from Leonardo and Fee (2005)

Aim of the thesis

My goal for this thesis is to explore dynamic neural field theory in the domain of sequence learning. Due to the available experience at the institute of neuroinformatics, the songbird suggested itself as a model animal that performs sequence learning on different levels.

As I envisioned to implement the model on neuromorphic hardware I imposed the hardware's constraints on any possible model: The model should use spiking neurons and Fusi synapses and it should not be too large.

Inspired by previous simple models of birdsong learning (Doya and Sejnowski 1995, Leonardo and Fee 2005, Hanuschkin, Ganguli et al. 2013), I developed a WTA-based spiking neural network model that is able to learn a song template and map it to the motor space using auditory feedback (see Fig. 14).

I explicitly use a modular code architecture in order to allow extensions to the model and reusability of the building blocks of the model in other projects.

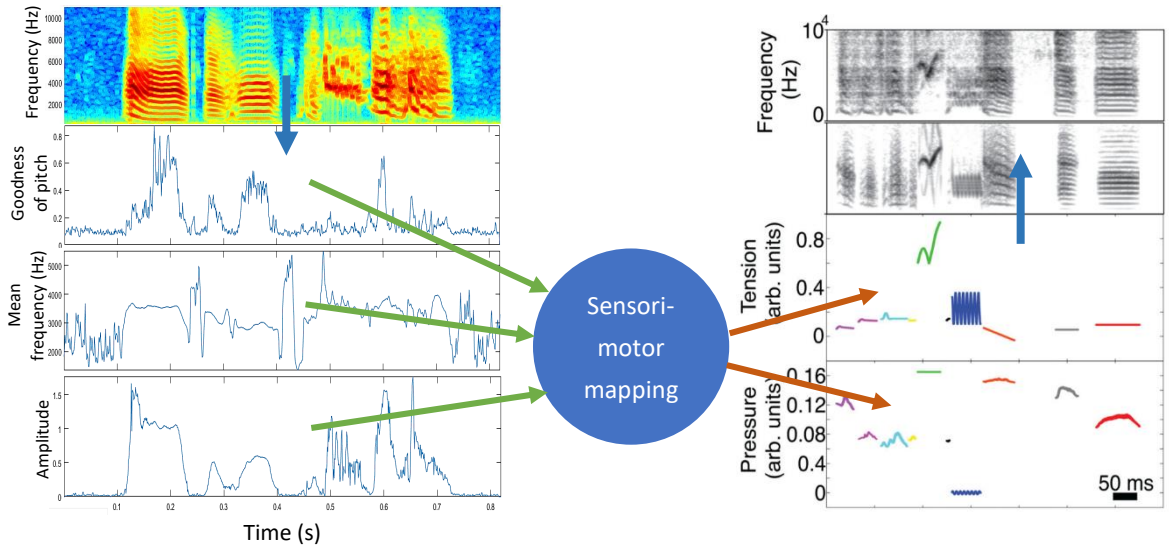

Fig. 14 The task, that is performed by the model is basically a form of sequence to sequence mapping. A trajectory through a perceptual feature space is mapped to a trajectory in a motor space. Here, a 2 dimensional motor representation based on air pressure and muscle tension is used. Right part of the figure from (Amador, Perl et al. 2013).

Methods

Before I present the whole model architecture for birdsong learning, I will describe the different building blocks of the model starting from neuronal level, going up to small circuits and finally more complex structures that might correspond to brain regions.

Building Blocks

Neuron Equation

For the project, I use the exponential adaptive integrate and fire model introduced by Brette and Gerstner (2005), as it is a simple but powerful neuron model and it is equivalent to the equation implemented in current neuromorphic AVLSI circuits (Livi and Indiveri 2009, Chicca, Stefanini et al. 2014). However, the model presented in this thesis is not dependent on the properties of the neuron equation and should work too with a simple integrate and fire model.

The membrane potential V of the exponential adaptive integrate and fire model is modelled with the following differential equation:

$$C\frac{dV}{dt} = -g_L(V - E_L) + g_L\Delta_T \exp\left(\frac{V - V_T}{\Delta_T}\right) - w + I_{syn} \quad (4)$$

w is the adaptation current, described by

$$\tau_w \frac{dw}{dt} = a(V - E_L) - w \quad (5)$$

I_{syn} is the sum of excitatory and inhibitory synaptic currents given by

$$I_{syn} = -g_e(t)(V - E_e) - g_i(t)(V - E_i) \quad (6)$$

At spike time, when V reaches the threshold V_T, V gets reset to E_L and a constant b is added to the adaptation current:

$$When\ V > V_{thr}:\ V \rightarrow E_L;\quad w \rightarrow w + b \quad (7)$$

The parameters are given in Table 1.

Table 1 default parameters for the neuron equation

Parameter	Value
C (membrane capacitance)	281 pF
g_L (leak conductance)	30 nS
E_L (leak reversal potential)	-70.6 mV
V_T (spike threshold)	-50.4 mV
τ_T (slope factor)	2 mV
τ_w (adaptation time constant)	144 ms
a (subthreshold adaptation)	4 nS
b (spike-triggered adaptation)	0.0805 nA

Fusi synapse

The Fusi synapse, introduced by Brader, Senn et al. (2007) used for all plastic synapses in this project is a bistable synapse implementing a form of spike timing dependent plasticity (STDP). It is implementable in neuromorphic AVLSI circuits as information is processed locally in space and time, so no explicit memory of spike timing has to be retained and as bistability allows efficient weight memory retention without using floating gates or memristors.

Each synapse has a weight variable X that is restricted to values between 0 and 1 and is scaled by a constant in order to obtain the synaptic weight. X is bistable, as there is a drift towards 0 or 1 depending on whether X is above or below Θ_x.

$$\frac{dX}{dt} = \alpha \quad if \quad X > \Theta_x \tag{8}$$

$$\frac{dX}{dt} = -\beta \quad if \quad X \leq \Theta_x \tag{9}$$

When a presynaptic spike happens at t_{pre}, and postsynaptic membrane potential V as well as a postsynaptic Calcium variable are in a predefined band, X is either increased or decreased by a constant a or b.

$$X \rightarrow X + a \quad if\ V(t_{pre}) > \Theta_V \quad and \quad \Theta_{up,l} < C(t_{pre}) < \Theta_{up,h} \tag{10}$$

$$X \rightarrow X - a \quad if\ V(t_{pre}) \leq \Theta_V \quad and \quad \Theta_{down,l} < C(t_{pre}) < \Theta_{down,h} \tag{11}$$

The postsynaptic Calcium variable is a function of postsynaptic activity with a long time constant:

$$\frac{dC(t)}{dt} = -\frac{1}{\tau_C} C(t) + J_C \sum_i \delta(t - t_i) \tag{12}$$

J_C is a constant value that is added to the Calcium variable at every postsynaptic spike.

See Fig. 15 for a visualization of the synapse working principle. Please find the parameter values used for the different Fusi synapses in the code.

Other, non-plastic synapses are simple synapses with an instantaneous rise, exponential decay kernel and as described e.g. by Roth and van Rossum (2009).

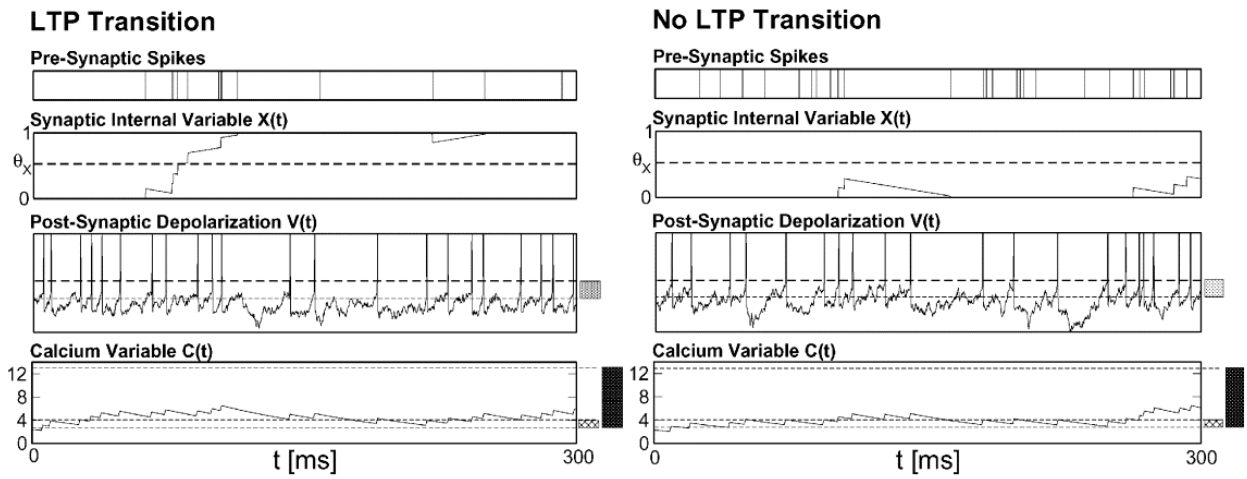

Fig. 15 The internal synaptic weight variable X increases, when both the Calcium variable c and post-synaptic depolarisation are above a threshold. Below Θ_x, there is a drift towards 0 and above towards 1.

Syrinx model

In order to generate motor output of the model that can be used as feedback for sensorimotor learning, we use a simplified model of the zebra finch vocal organ, the syrinx. The syrinx is located in the trachea of the birds. When air flows through, tissues called labia vibrate and generate a sound. By changing the tension of membranes with muscles, the sound can be changed. Based on seminal work by Titze (1988), who introduced a model of air flow induced oscillations in the human voice system, Perl, Arneodo et al. (2011) presented a simplified model of the bird's syrinx. The following two differential equations describe the oscillation of the labia:

$$\frac{dx}{dt} = y \tag{13}$$

$$\frac{dy}{dt} = \alpha\gamma^2 + \beta\gamma^2 - \gamma^2 x^3 - \gamma x^2 y + \gamma^2 x^2 - \gamma x y \tag{14}$$

There are 2 parameters α and β that are able to change the sound. They can be interpreted as muscle activity and air pressure.

Winner-take-all networks

Dynamic neural fields, as described above, can be discretized and implemented as winner-take-all networks. Winner-take-all networks can be implemented as spiking neural networks.

A winner-take-all network is modelled as two groups of neurons: An excitatory population that is ordered into a field, so every neuron has an x position and for 2 dimensional fields also a y position. And an inhibitory population that receives synapses from all excitatory neurons and also inhibits the whole excitatory population.

Additional to the global inhibitory connections, there are local excitatory connections. Every excitatory neuron is connected to its neighbours with a strength dependent on its distance to the respective neighbour. No synapses are created above a certain cut-off distance, that was chosen to be 8. The strength of every connection in the 1 dimensional field is a simple Gaussian, in the 2

dimensional field, it is given by the following kernel function multiplied by an independent parameter g_{ex}:

$$synapse\ strength = \left(1 - \frac{x^2+y^2}{2*\sigma^2}\right) * Exp\left(-\frac{x^2+y^2}{2*\sigma^2}\right) * g_{ex} \quad (15)$$

This is negative Laplacian of Gaussian kernel; the shape is also called a Mexican hat. Sigma is the parameter that determines the broadness of the kernel. Due to the global inhibition, a Mexican hat shaped kernel does not seem necessary but it turned out that fields are much more stable and easier to tune than with a Gaussian kernel, as strong activation peaks are less prone to explode and take over the whole field irrespective of global inhibition, which cannot be mad arbitrarily large in order to still allow self-excited peaks.

In the simulations shown in the results section, 2 dimensional fields are of squared size with 16x16 neurons.

Synfire Chain

It has been shown that chains can emerge using STDP or similar learning rules in a biologically plausible way (Li and Greenside 2006, Jin, Ramazanoğlu et al. 2007, Gibb, Gentner et al. 2009, Fiete, Senn et al. 2010, Long, Jin et al. 2010, Brea, Senn et al. 2013). I do not model the formation of a synfire chain, but use group of neurons with a predefined sequential connectivity.

Sequence model

As the higher level sequence model, representing the syllable sequence, the DNF-based sequence model described in the introduction is used. As CoS, either a special silence detector is used in order to recognize the silence between syllables or a certain location in the sensory field that stands for silence (without dimensionality reduction: low amplitude) activates the CoS node. Given a teacher signal, CoS can also be plastically connected to the last chain neuron of a syllable (which would include the silence).

Self-organizing map

Inspired by topology preserving feature maps in the brain, Kohonen (1982) introduced a self-organizing map (SOM) algorithm. Using competitive learning, the algorithm reduces dimensionality of a high dimensional input vector while preserving topological properties of the input space.

The classical implementation of an SOM model consists of a one or two dimensional field of neurons and an associated, usually randomly initialized, weight vector with the dimensionality of the input vectors. Classically the algorithm has the following steps:

1. Select input vector randomly from the set of inputs.
2. By calculating the Euclidean distance from the input vector to all the weight vectors, find unit in the field that matches input vector best (best matching unit, BMU)
3. Adjust weights in a certain radius (decreasing with time) around the BMU to fit the input vector better.
4. Repeat from 1.

The algorithm has been implemented in spiking networks by Ruf and Schmitt (1997) using spike timing instead of rate based coding, allowing a fast and local competition. In this implementation, every neuron, modelled as integrate and fire neuron, receives a weighted sum of the input vector. Assuming that both input and weight vectors are normalized, this weighted sum is the angle between the two vectors, so a neuron fires earlier, the closer its weight vector is to the current input vector. Competitive learning is implemented by lateral excitation and global inhibition in the field of neurons (a winner-takes-all network). Ruf and Schmitt (1997) suggest an instar learning rule where the weights are updated proportional to the difference between input and current weight value and stronger if the neuron spikes earlier:

$$\Delta w_{ij} = \eta \frac{T_{out} - t_j}{T_{out}} (s_i - w_{ij}) \tag{16}$$

s is the current input vector, w the weight vector, t the spike time, i is the index for the input and j for the neurons, η the learning rate and T_{out} is the time after which no learning happens. This latter global reference time makes the model unsuitable for implementation in biological or silicon hardware. Furthermore, the network has to be reset after every learning step and continuous input is not possible. Rumbell, Denham et al. (2014) address those issues by a spiking SOM model that only uses an STDP learning rule and structures its input automatically via oscillations of the input population.

The spiking SOM that was implemented in this project is based on the model of Rumbell, Denham et al. (2014). The map is a 2 dimensional winner-take-all field as described before. Input to the field is provided as 1 dimensional fields, on which feature values are encoded as a local peak of activation on the field. Feature values are scaled to the interval between 0 and 1 and are then fed as Gaussian input peak into the 1 dimensional field (e.g. the centre of a peak coding for a value scaled to 0.5 would be exactly in the middle of the field), which results in early and strong firing at the centre of the peak and late and weak firing in more remote regions. All input fields share a common inhibitory population that is tuned so that it strongly inhibits activity in the input fields but takes some time to be activated leading to an oscillatory behaviour in the input fields automatically structuring the input into samples. Input and self-organizing map are connected with randomly initialized all-to-all plastic synapses. When input spikes arrive, the synapses to the early firing neurons to the responding neurons in the map are strengthened by STDP. The first neuron that responds in the map is the BMU, as it is most strongly activated and therefor reaches its firing threshold first. The BMU activates neurons in its close neighbourhood through lateral excitation allowing them to form synapses to the input too and at the same time inhibits other regions in the map that might also be close to the input. The lateral connectivity kernel of the map shrinks with time.

Neural dynamic architecture for sensorimotor learning

The spiking dynamic neural field architecture developed in this thesis consists of six parts (see Fig. 16): A self-organizing map that receives feature values as an input, perceptual motor fields, a serial order sequence model, a synfire chain and a syrinx model as motor output. The building blocks have been introduced on the previous pages.

A tutor song motif, that is the input to the model consists of a couple of syllables. From such a motif, features, like amplitude and mean frequency are extracted. Features are extracted from small overlapping bins of the spectrogram in order to obtain a trajectory of features without large jumps. In a first phase of learning, the extracted feature values are fed into the SOM that maps the feature space into 2 dimensions, so a song motif is represented as a trajectory in the 2 dimensional perceptual space. In the second phase of learning, the memory of the tutor song template is acquired. The synapses between the perceptual field and the sequence and chain neurons are gated off, in order not to interfere with the learning, so they learn silently. At the start of each rendition of the tutor song, the sequence is triggered and perceptual song motif trajectory and chain sequence run simultaneously and get connected by plastic Fusi synapses. When the song template is acquired, a random walk is induced on the motor field. The output of the motor field, basically the projection of its activity on its two axes, drives the syrinx model that produces a sound. This sound is fed back into the model and elicits, only if it hits the right note, after some delay, an activation in the chain neurons (whose chaining is gated off temporarily). Thanks to this feedback, chain neurons, that basically code for a time point in a syllable, can be paired with their correct motor counterparts. In the final production phase, the sequence model is triggered manually so that it replays the learned sequence. The chain neurons now drive the correct motor trajectory that reproduces the syllables of the song template. In Fig. 18, the model is explained in more detail.

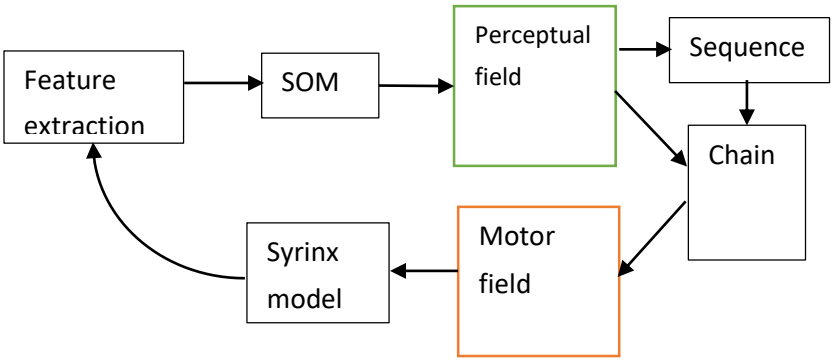

Fig. 16 Schematic overview of the architecture presented in this thesis. Features are extracted and fed into the SOM which reduces dimensionality to two. The song is represented as a trajectory in the perceptual field. This trajectory is learned hierarchically. The song motif is made up of short syllables. The trajectory of the syllables is learned by linking locations in the perceptual space to a sequentially active chain of neurons and the order of the syllables is learned by a DNF based serial order working memory model. After learning by feedback, the chain can elicit a trajectory in the motor field that corresponds to the stored song motif. The motor field drives a model of the bird's syrinx which closes the feedback loop as the output of this model is again feature extracted and fed into the SOM.

In Fig. 19, an alternative model architecture is described, in which the chain neurons do not play the role as intermediary between the perceptual and the motor field. The sensorimotor map is learned directly between the two fields. The chain neurons drive a trajectory in the perceptual field in this model which then gets translated in the motor space by the sensorimotor map. Apart from this, the models are identical. In the following results section, only the results of the first model are shown in order to avoid confusion as the results are very similar.

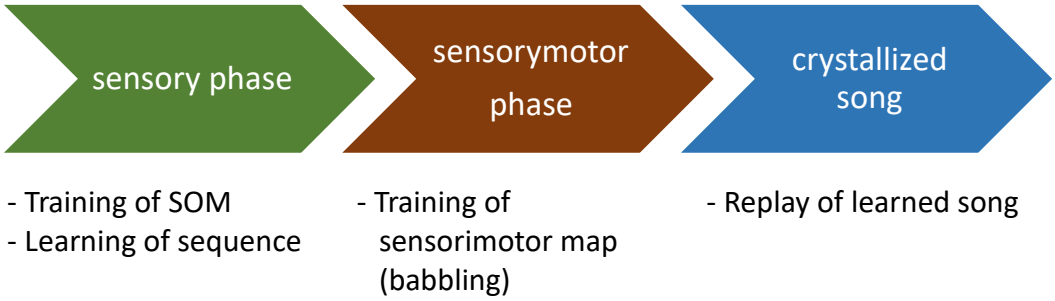

Fig. 17 Different phases of learning in the architecture. In the first phase, the SOM is trained and the song template is stored. In the second phase the sensorimotor map is trained, which means that the model learns which motor state has to be activated in order to produce a certain sensory state. Finally, the complete learned song is replayed and not changed anymore.

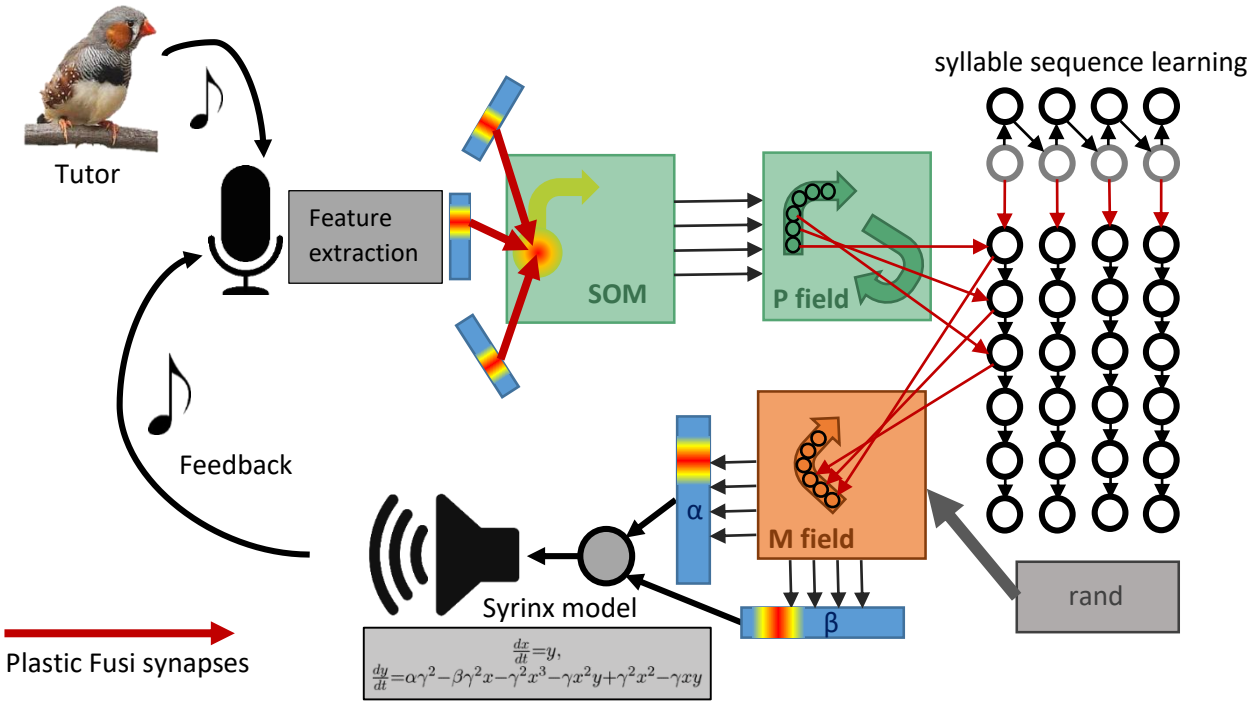

Fig. 18 Detailed schematic of the of the model architecture.

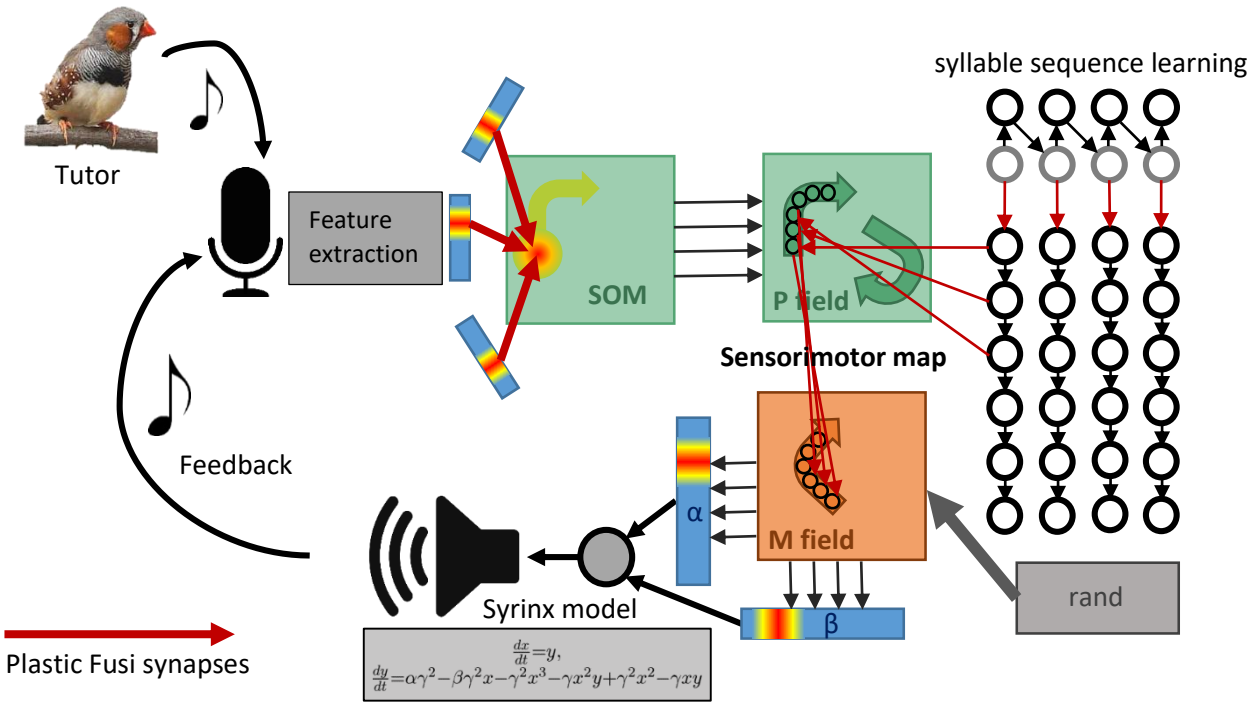

Fig. 19 Alternative model architecture. In this architecture, during the sensorimotor phase, a sensorimotor map from the perceptual to the motor field is built. Once the map is learned, a sequence that is driven in the perceptual field by the chain neurons, is directly mapped into the motor space. The song template is in this model stored in the synapses from the chain neurons to the perceptual map and not the other way round. The other parts of the architecture remain untouched.

Software implementation

All simulations were performed using the Python based framework Brian 2 as part of the anaconda distribution (4.2.12) with Python 3.5 on a win 64 platform with a 3.5 GHz Intel i3 processor and 16 GB RAM.

For feature extraction of the tutor song and of the feedback the Matlab toolkit Sound Analysis Tools (SAT) was used. It can be downloaded from http://soundanalysispro.com/matlab-sat. In order to use it with python based Brian 2, I made use of the MATLAB Engine API for Python that is part of Matlab's default installation from MATLAB R2014b or later.

All modules described were implemented as python functions made available in the groups GIT repository to allow a building-block like reusability and straightforward extensibility of the model.

Results

Before presenting the results of the song learning model, I will show characterizations of some modules used to build the model as they are necessary for the understanding and especially for the tuning of the model. All those modules were implemented in python as modules that can be used by others to build similar models.

Characterization of the different modules

Neuron equation

I start with the characterization of the neuron equation. In this model, most integrate and fire models would be suitable. As it is equivalent with the equation used in the chip, I used the equation described by Brette and Gerstner (2005). The basic behaviour of the equation is shown in Fig. 20. The form of a spike in the exponential integrate and fire model is characteristic as just before a spike is triggered the membrane potential increases exponentially.

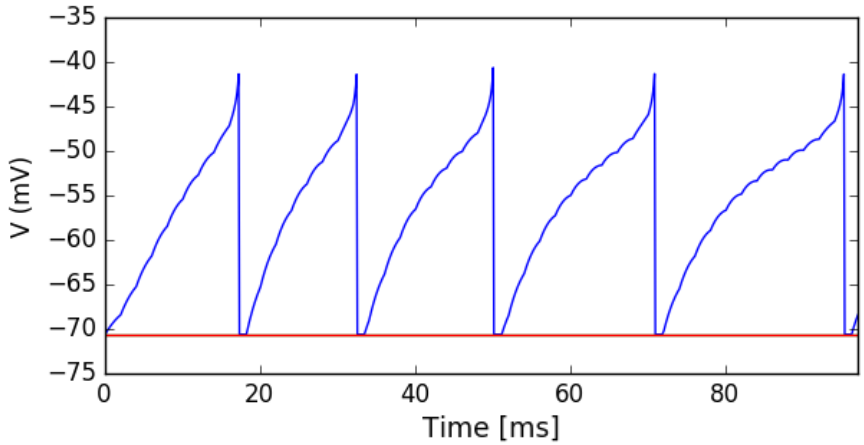

Fig. 20 Behaviour of the neuron equation. The neuron integrates a constant rate 500 Hz input. Resting potential (red) is at -70.6 mV. The distance between spikes increases despite constant input due to adaptation. When V_T=-50.4 mV is reached, the membrane potential rises exponentially to the reset threshold.

Synfire Chain

The synfire chain is a simple chain of neurons that are connected serially. When the first neuron gets activated by a short burst, a wave of activity travels through the chain. In Fig. 21, I show the chain for different values of the parameter that determines chaining strength. When this strength is higher, the chain gets steeper. If this is done during song replay, it means that the song becomes faster.

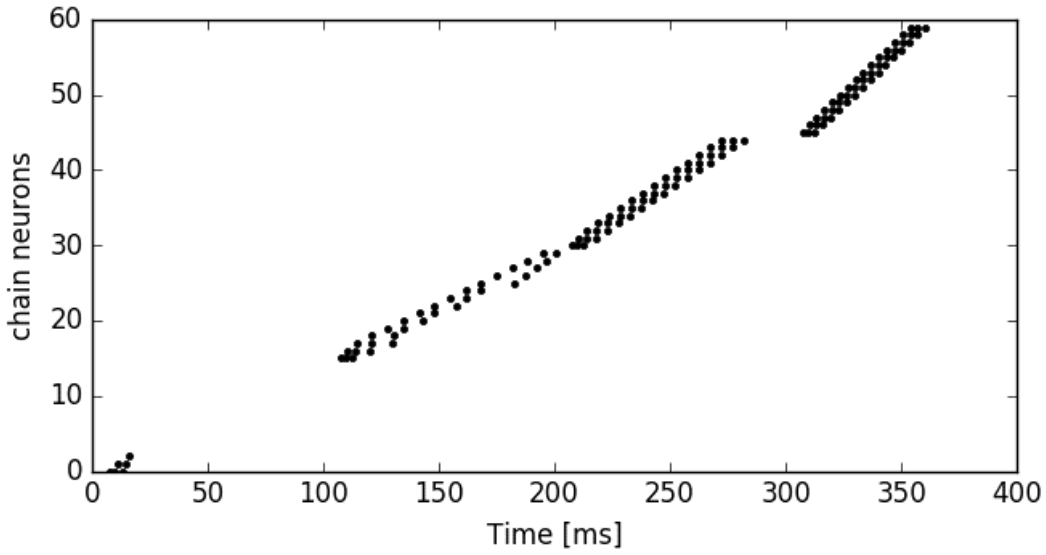

Fig. 21 Raster plot of the synfire chain for different values of the excitatory parameter. The chain gets steeper for stronger excitation (right) and ceases to exist for too low excitation (left). Each chain consists of 15 neurons.

Feature Extraction

Fig. 22 shows features extracted from a whole song motif. 4 different syllables can be clearly seen. In Fig. 23, I show the single syllable that is used in the following simulations as an example for a real bird song syllable.

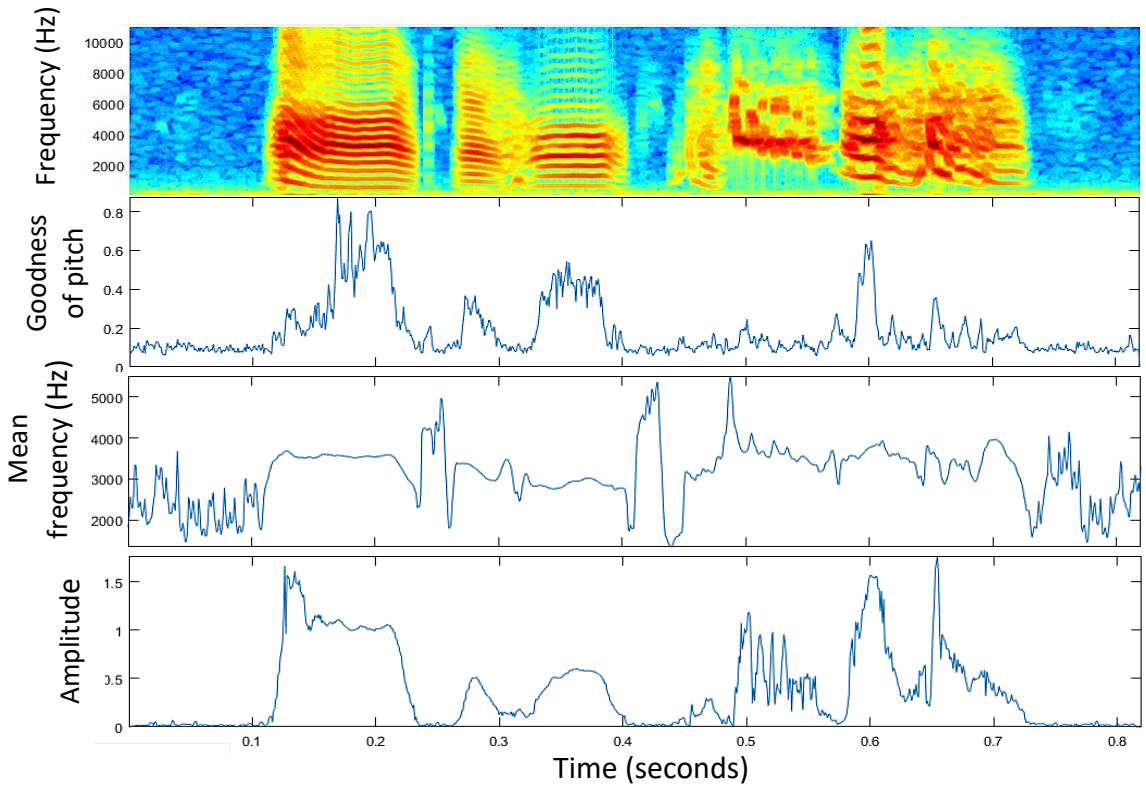

Fig. 22 Feature extraction for one song motif with 4 syllables.

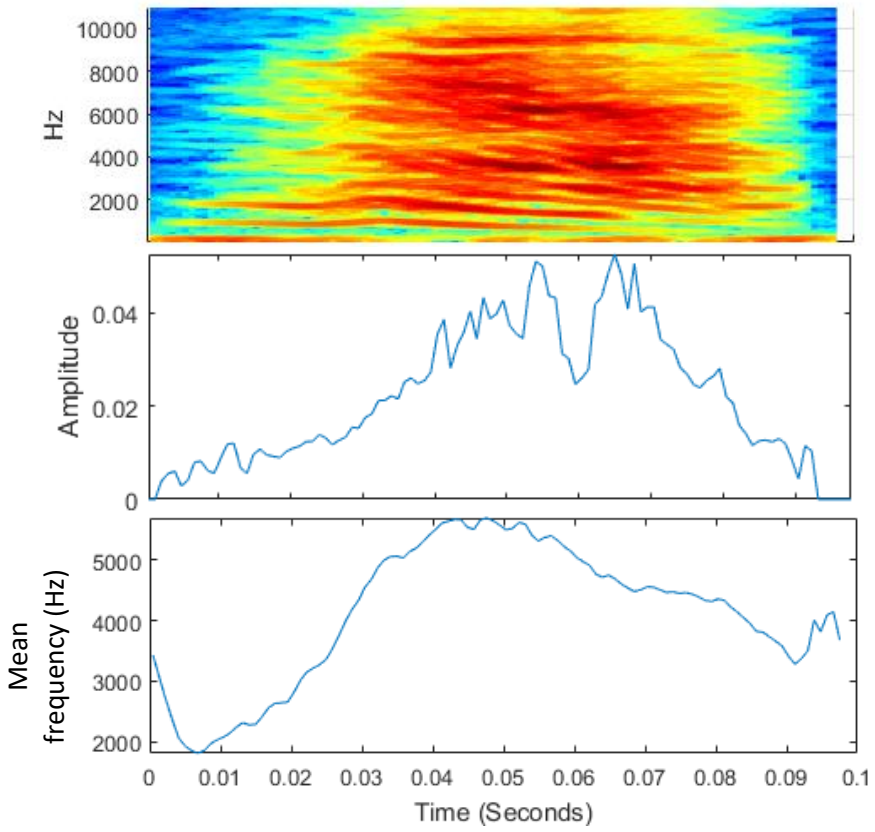

Fig. 23 Features of the real song syllable used in the following experiments as an example syllable. Only the two features Amplitude and mean frequency are shown, as they are used as input to the perceptual field

Self-organizing map

I implemented a spiking self-organizing map based on dynamic neural fields and tested it on colour learning, before going on to birdsong learning, as colours are easier to handle. However, the map turned out to be rather hard to tune. I finally achieved differential activation of the map and therefore the calculation of a best matching unit. But before the map can be actually used, the learning needs to be validated, which means, one would have to check if the neighbourhood relations of the input are still there in the mapped locations.

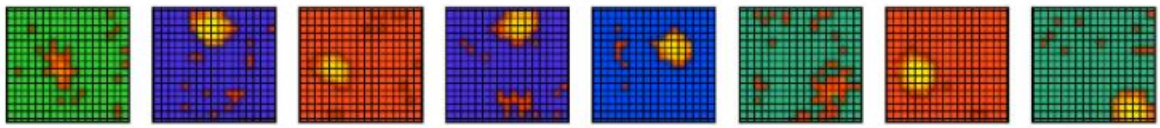

Fig. 24 Excerpt of the responses of a spiking self-organizing map trained on colours. We clearly observe differential activity of the map to different colours, so a BMU is successfully determined using the spiking mechanism. However, if the map is correctly learned and the neighbourhood relations are correctly retained, needs to be tested in a more in depth analysis.

Sequence learning module

The serial order model described earlier is part of the high level sequence learning in this project. In Fig. 25, I show a simulation of the model that acquires the order of 2 peaks in the content WTA in one shot and is able to replay that sequence later.

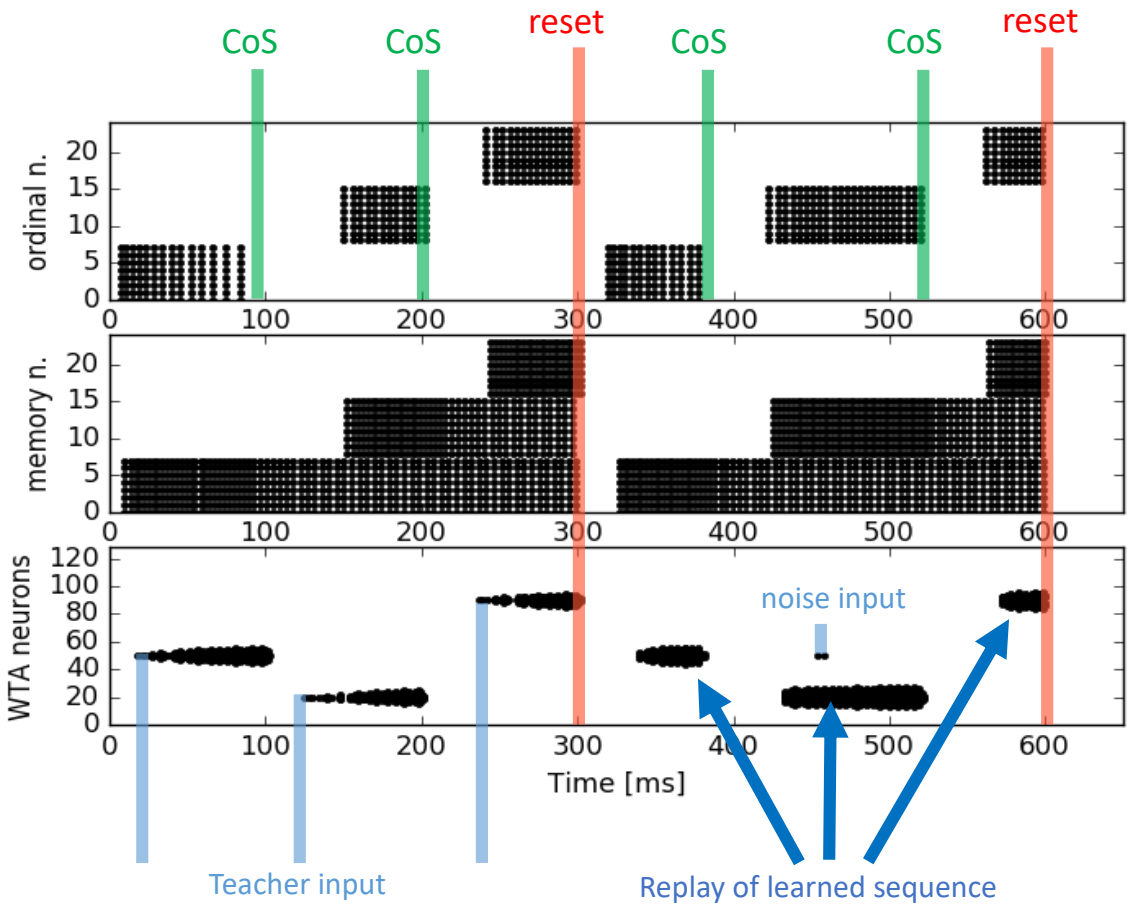

Fig. 25 Raster plot of brian 2 simulation of the DNF-based serial order model. It contains 3 groups of ordinal and memory neurons. In the first round (left) the system learns the synapses between ordinal group and WTA content field that is activated by a teacher signal. In the next round, the content is replayed by the model. CoS and reset signals are hard coded and shown in green and red in this demo.

Neural dynamic architecture for sensorimotor learning

So far, I can only show simulation results of the model's core. The SOM and the syrinx model are circumvented and feedback is fed directly into the perceptual field, so the sensorimotor mapping is a 1:1 mapping between the two fields. Any other linear mapping could of course be used too, but this was chosen so that one can immediately see in the resulting plots that the model is able to learn the trajectories (as they are basically the same). As a tutor song I used for illustration purposes an artificial trajectory of activations from the lower left corner to the upper right corner and a trajectory of song features extracted from a real song, as shown in Fig. 22. Only the 2 features amplitude and mean frequency were used.

In order to make clear which plot refers to which phase of learning, the colours from Fig. 17 are used. Green for the sensory phase, in which the song template is acquired, orange for the sensorimotor phase, where the sensorimotor map is learned and blue for the crystallized phase, where the song is replayed. Furthermore, there is yellow for a phase, where the system only hears the tutor song again after it has already been learned. This is interesting, as mirror neurons fire there and syllables can be recognized in by the sequence model (see Fig. 26).

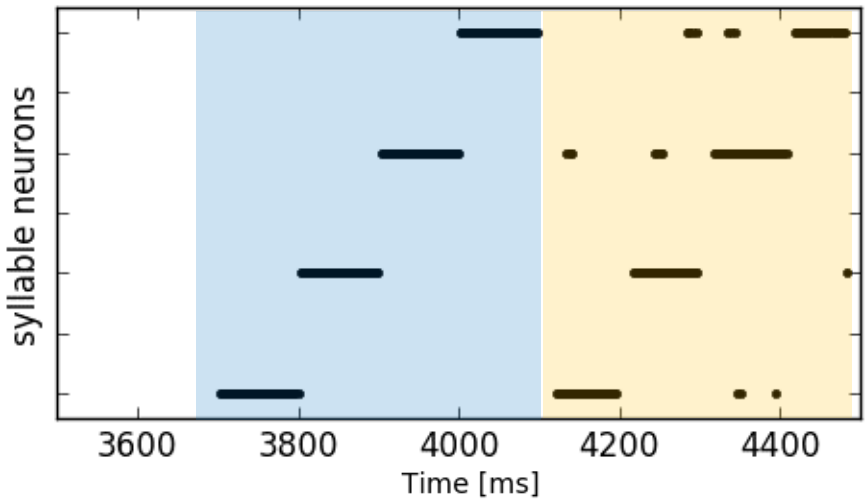

Fig. 26 Syllable recognition. The system can not only play a song syllable, but is also able to recognize its own song when it hears it. The left part of the figure is the activity of the sequence neurons during song replay (blue) and the right part is the activity during listening to the own song (yellow). A single sequence neuron is shown. The song consists of 4 syllables of about 100 ms each.

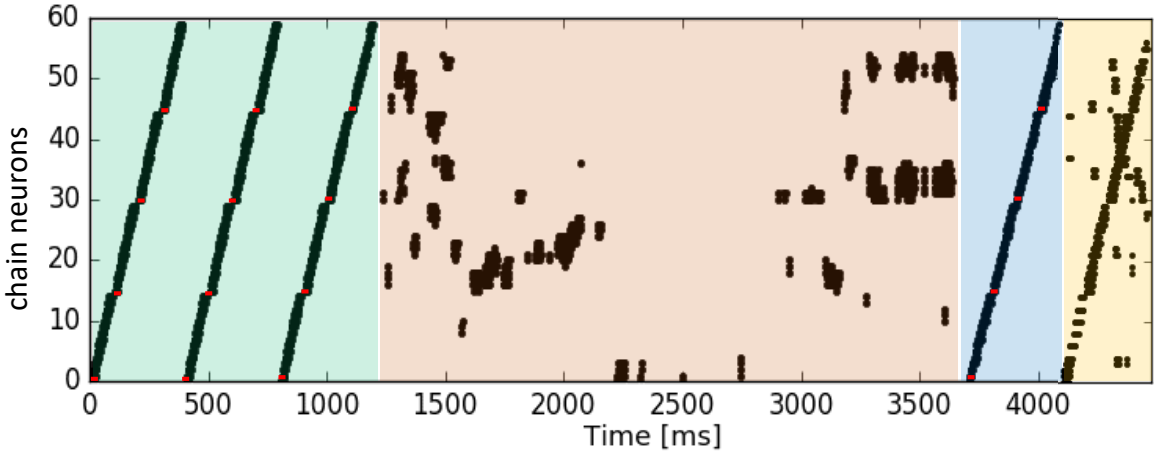

Fig. 27 Raster plot of chain neurons during the whole experiment. Waves of activity are triggered in the sensory phase (green) and during replay of the song (blue). In the sensorimotor phase (here shorter than usual for easier plotting), chain neurons are activated by the feedback to the random walk. During hearing (yellow), when they are not gated off, chain neurons respond to the tutor song. Like mirror neurons they have both auditory and motor capabilities. The chain trigger input is depicted as small red bars. The song in this experiment consists of 4 syllables. For simplicity, only 2 of them are shown in later figures.

In the following figures Fig. 28-Fig. 30, I show the activations of the sensory and the motor field throughout the experiment. Fig. 28 shows the spiking activity of the sensory field in the sensory phase, Fig. 29 the random walk activity of the motor field in the sensorimotor phase. In Fig. 30, I show the motor field during the replay of the song. In the complete model, this activity would be used to drive the syrinx model and reproduce the tutor song. Fig. 31 and Fig. 32 show the weight matrices of the incoming and outgoing synapses to and from the chain neurons. Each weight matrix is a snapshot of the activity in the respective field from the point in time at which the chain neuron is active.

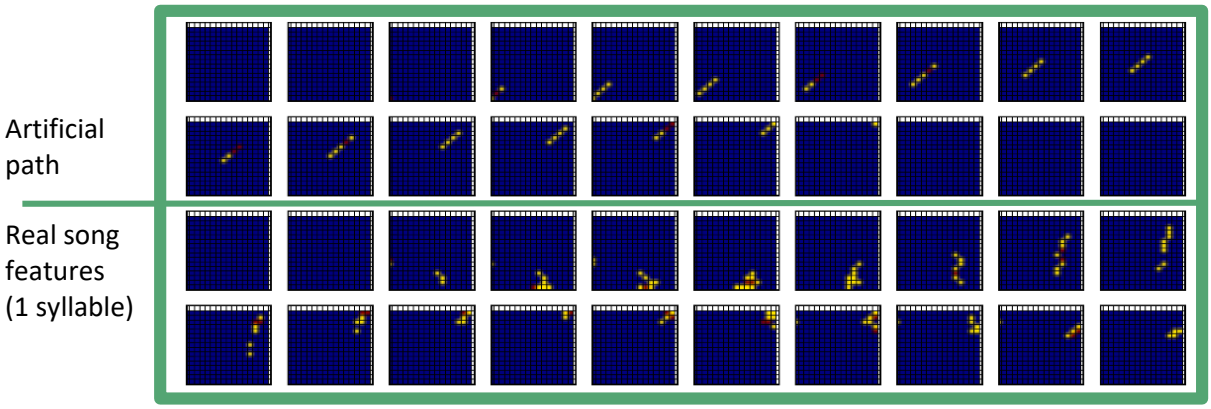

Fig. 28 Sensory field in sensory phase, spike count over 5 ms per tile, two syllables are shown, one artificial trajectory and one trajectory of real sound features, as shown in Fig. 23.

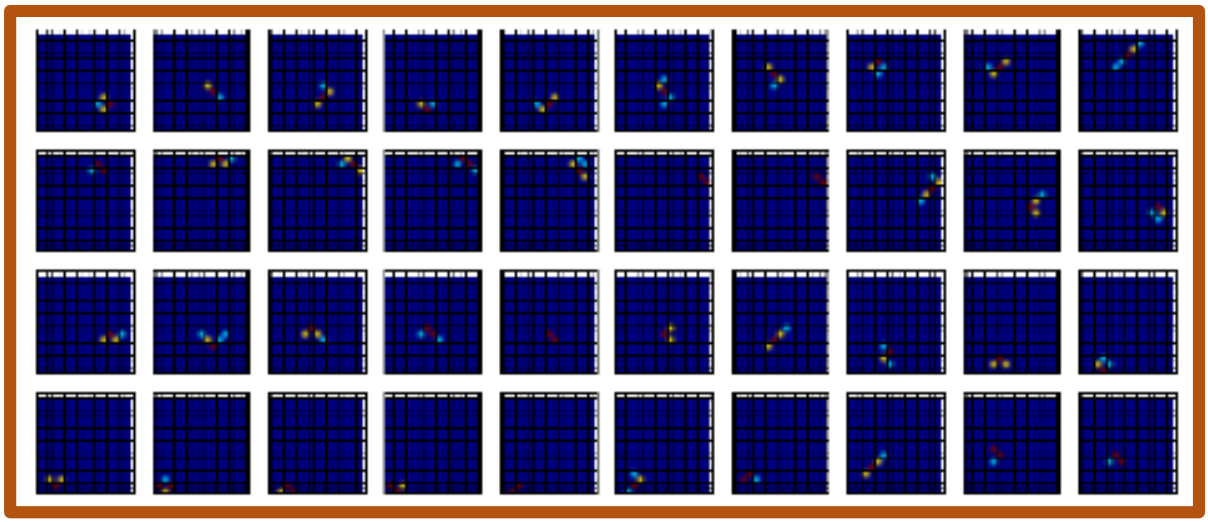

Fig. 29 Motor field in sensorimotor phase, spike count over 5 ms per tile. Random walk activity of motor map during the sensorimotor phase. In order to sample the whole space, the random walk is much longer, here only 200 ms are shown.

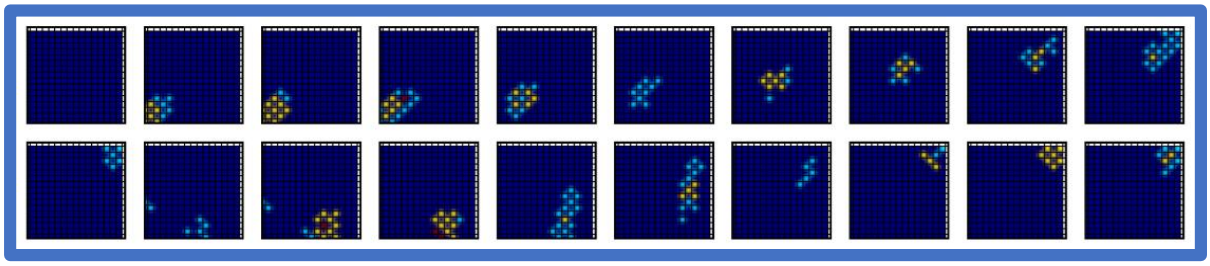

Fig. 30 Motor field during song replay, spike count over 10 ms per tile. A 1:1 mapping is used as an example to show learning, instead of mapping with the syrinx model. Therefore, the trajectory in the motor field looks very similar to the trajectory in the sensory field.

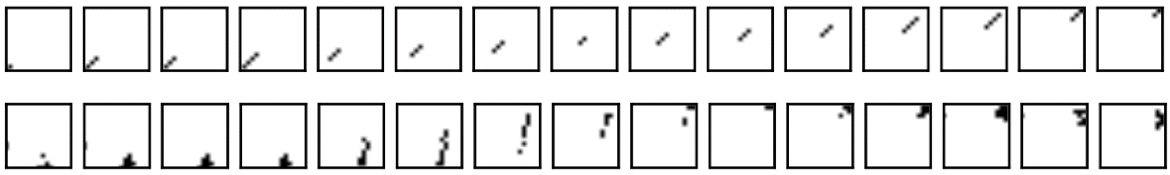

Fig. 31 Weight Matrices from sensory map to all chain neurons, one tile depicts one chain neuron and is therefore a snapshot of the field's activity at the respective position in the chain.

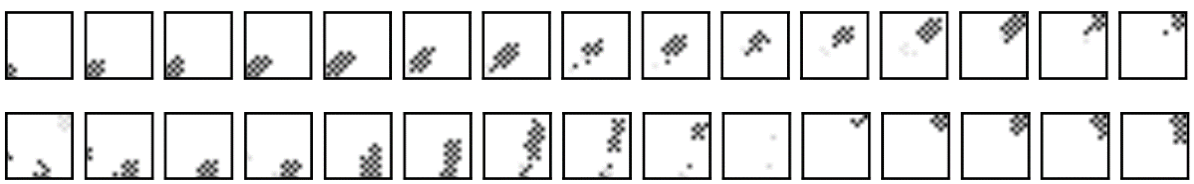

Fig. 32 Weight Matrices from all chain neurons to sensory map, one tile depicts one chain neuron.

Discussion

In this thesis I present a songbird-inspired spiking neural network model for sensorimotor sequence learning based on dynamic neural fields that can be implemented as winner-take-all networks on a neuromorphic chip and I show in the results that the core of the model works and is able to learn a simple sensorimotor mapping and real song trajectories. The model allows very fast learning due to its simple structure. It also allows us to understand its working principle easily due to its modular structure and sparse activity. Although the model does not provide a plausible mechanism for song learning in zebra finches, I will discuss analogies of the model to biological systems and propose extensions to the model that capture song learning in a better way. Furthermore, I will discuss drawbacks and strengths of the model and the chosen modelling approach from different points of view. Finally, I suggest to implement the model into a neuromorphic system for vocal imitation learning.

Biological analogies

Although the model is not primarily designed to fit song learning, it can still offer some biological insights as it is designed with similar constraints in mind (e.g. local computations, spiking neurons). The mechanism, how the sensorimotor map is acquired by the bird is different from the "brute force" random walk sampling method I use, but still, the model offers a plausible mechanism how the song template is stored and how a chain, as found in HVC is able to drive a sequential motor behaviour in a spiking network.

Time cells

Such chains, also called "time cells" (Markowitz, Liberti III et al. 2015), cannot only be found in the bird's HVC but also in other species mainly in hippocampus and cortex in various behavioural contexts (Pastalkova, Itskov et al. 2008, MacDonald, Lepage et al. 2011, Harvey, Coen et al. 2012, Markowitz, Liberti III et al. 2015).

In this thesis I present to my knowledge the first simulation of a spiking neural network model that links sequential "time" cell activity to attractor dynamics in DFT.

Mirror neurons

The same chain neurons that form the "time cells" also function as mirror neurons in the presented model. In line with the definition by Rizzolatti and Craighero (2004), the neurons are triggered by auditory stimuli, whereas they are also able to elicit those exact sounds to which they respond with the vocal motor system. Those neurons and are therefore the essential units for imitation learning.

The model can be seen as a minimal model of sequence imitation learning. In the songbird, the situation seems to be a little bit more complex, as HVC_X-projecting neurons and not HVC_{RA} mirror auditory activity, especially when they hear the bird's own song (BOS) (Prather, Peters et al. 2008).

Please note, that mirror neurons are easier to achieve in the auditory domain than in the visual, as the own song and a foreign song are already in the same space. A hand movement from someone else however has still to be mapped to the own coordinate system. This insight gives us an

information about a possible applicability of the presented model in the visual-motor domain: We would have to perform the necessary transformations first (which could be done using concepts of DFT), if we would like to apply this model.

Imitation learning is also essential for human learning (speech, language, dancing, driving, etc.) and useful for robotics.

HVC and RA

If I would have to map the modules of the model to regions of the bird's brain, I would say that the chain neurons are most similar to HVC and the motor field is analogous to RA. The perceptual field is outside the classical song system in an auditory region like NIf. As already discussed, the mechanism of learning of the connections between chain and motor field is not plausible, also because HVC_{RA}-neurons do not respond to auditory feedback. The learning paradigm that is currently most discussed for these synapses is reinforcement learning driven by variability in LMAN (Fee and Goldberg 2011). By taking the firing of the previously described mirror neurons (a.k.a. chain neurons) as a reward signal and by adding an additional field with self-sustained activity parallel to the motor field, that stores the rewarded regions, I could extend the model easily with a greedier sampling mechanism.

In addition to the error/reward signal provided by the chain neurons, an error signal can easily be computed using the DNF based representations. By duplicating the auditory field and learning the synapses from chain to auditory field, we have a forward model that allows us to compute a difference between the two fields (a difference between forward model prediction and actual feedback input).

Trajectory encoding in a neuron population

From a computational point of view, the sequence learning core of the model can be compared with a state machine. The meaning of the states is bound to them by learning.

This analogy is of course not a coincidence as from a computational point of view, there are basically only 2 different straightforward ways to encode temporal sequences in a population of neurons. Here, I discuss those and state why I decided to structure the model like I did after considering the alternative.

Let us assume, there is a group of neurons that uses space coding for values on one or two dimensions, like e.g. the motor field described in the model. I we want to encode a sequence in those neurons, the first thing that comes to the mind is, to connect the neurons of the sequence. However, obviously, there is a problem, when one neuron appears twice in such a sequence as we can no longer know which way we should go on after this neuron as there are two outgoing connections. We would need an additional state to remember if we are at this neuron for the first or the second time (see Fig. 33).

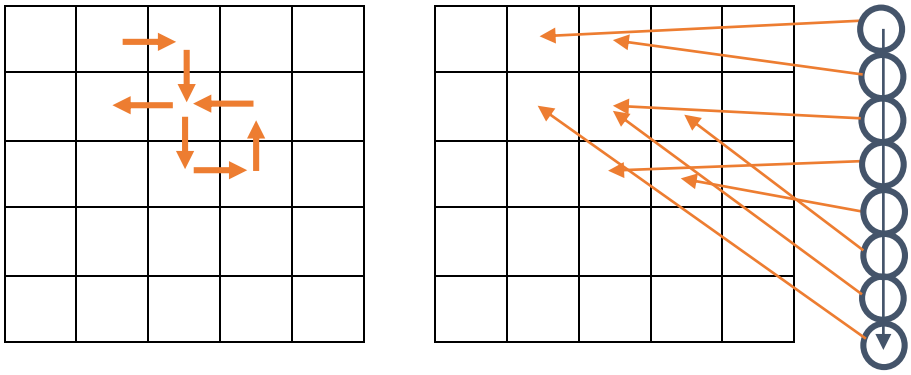

Fig. 33 Illustration of the problem that additional states are needed for unambiguous sequence representations. On the left, a sequence with a loop in a 2 dimensional field is shown. If the field should be kept as it is, the only way to represent a sequence unambiguously is to add additional states, a solution to this is depicted on the right.

Obviously, no model (not even the brain) can get around this well-known constraint. As, in a static population of neurons, we cannot just create a new state/ a new neuron whenever a loop of the sequence happens there are basically 2 ways to circumvent the problem: Either we add an additional system with a sufficient amount of states to remember which ways have already been taken or we avoid loops by design of the field.

If we choose the former approach, it is obvious that in order to remember all possible ways, the system has to have the same number as states as the number of elements in the sequence, as in an extreme case the sequence could completely consist of one single neuron.

If we choose the latter approach, we would have to find a way to incorporate the flow of time into the spatial field. The easiest way would be to just copy the whole field for every time step and create an additional time dimension like that. This seems however not very efficient, as we create much more states than actually necessary (as stated before, only one state per time step is necessary in the worst case). However, in order to avoid this (if we do not allow ad hoc neuron generation, which would be possible in the brain especially in hippocampus), we would have to use some prior information of what kind of sequences can be expected. We could e.g. save a lot of states if we knew the maximum number of repetitions of every neuron in the sequence or if we knew that only loops of less than 10 elements can happen (we would only have to keep track of those). The embedding of time information however, can also be learned. A SOM that uses the relative time as a feature could be used to learn such an embedding with perfect memory. Alternative SOM-algorithms like TSOM or RSOM with a restricted memory have been developed (Chappell and Taylor 1993, James and Miikkulainen 1995, Chappelier and Grumbach 1996, Euliano and Principe 1996, Varstal, Millán et al. 1997).

If we give up the constraint, that only one neuron is active at any given time point, the problem gets more complicated but now there are so many possible combinations that a repetition of the same combination is unlikely as long as they are randomly chosen. Unlike in DFT, where spatial or population coding is used, combinatorial coding would have to be used to assign a meaning to the states (like e.g. in the olfactory system (Malnic, Hirono et al. 1999)). However, note that when we

assign meaning to the nodes, the problem of loops arises again and we still would have to look for an embedding or external coding of temporal information.

An advantage of an embedding of temporal information into a population is that recurrent connectivity can be learned and used to generate sequences (as e.g. proposed by Nowotny, Rabinovich et al. 2003).

If we add an external one dimensional representation of time, an important advantage is that the same real world state (as part of a sequence) activates the same perceptual state when it reoccurs in a sequence. Therefore, I decided to use this approach as it is most compatible with DFT and most interpretable (as already discussed, mirror neurons and "time" neurons emerge).

Please note, that such a chain of neurons does not need be prewired and spatially coded as in this model but it might arise ad hoc from an unordered network and be activated by context.

If the brain uses a such spatial coding scheme at all, it seems probable that both proposed mechanisms are used one way or another, as sequences are ubiquitous and happen in many regions and many levels of hierarchy.

Discussion of the approach

In this part of the discussion, I try to assess strengths and drawbacks of the project from different points of view.

Advantages of the DFT approach

The main advantage of the model, in my opinion, is its simplicity. The working principle is straightforward and all plastic synapses are trained with STDP, which means, that no complex learning rules or backpropagation are used.

The reason why it can be so simple and intuitive is mainly because it uses dynamic neural fields, as they solve some problems intrinsically: In order to learn the sensorimotor map, the system has to keep a working memory of the past motor activity, as the feedback takes some time (in the range of a couple of milliseconds) to be processed. As dynamic fields work as a low pass filter of the activity, this might already be enough (for short delays, like 2 ms in the simulations, it was sufficient). If the delay gets longer, an additional memory field can be added to the motor field and store past activity. An implementation of this memory purely by an eligibility trace as proposed e.g. by Hanuschkin, Ganguli et al. (2013), is not possible, when only Fusi synapses are used as additional to the Calcium threshold there is a threshold for the postsynaptic membrane potential.

Another strength of the WTA fields is that they can automatically find a good path through the motor field, even if the sensorimotor mapping is not unique i.e. if there are several locations in the motor field that produce the same sound. In such a case, due to global inhibition, the system will choose only one possibility and due to local excitation, it will choose the closer location.

There is however one potential caveat with a model-guided approach: Although DFT is very powerful it can probably not be used as a universal hammer to destroy all problems, so it is always important to carefully think about where the usage of a framework makes sense and where not.

Drawbacks

Notwithstanding, the model also has some drawbacks: Although the model is simple, the spiking model implementation make it hard to tune as there are many parameters and as there are no ready-to-use modules or examples. This slows down the process of testing new ideas dramatically. This is why, during the thesis I developed python modules, based on Brian 2 that allow a modular structure on a higher level than offered by Brian 2 itself. Together with a documentation, and default parameters that I will provide in the code and with contributions from other group members, this allows to easily build larger WTA based architectures in the future.

Another problem, that is still unsolved, is the fact that real songs might be too dissimilar to what the syrinx model can produce, therefore, I so far did not achieve a good sensorimotor mapping from a real song sensory field to the motor field connected with the syrinx model.

Machine learning perspective

When assessing a model, it is usually a good idea to look at what related fields offer. From the point of view of machine learning, the model seems not particularly interesting, as it solves a problem, that can easily be solved better by other methods. The state of the art machine learning technique for sequence to sequence mapping are recurrent neural networks (Sutskever, Vinyals et al. 2014) which since recently allow image captioning (Vinyals, Toshev et al. 2015), speech recognition (Graves, Mohamed et al. 2013) and generation (Zen and Sak 2015) and language translation (Bahdanau, Cho et al. 2014). But as the sensorimotor mapping is not even a sequence, a simple perceptron (Rosenblatt 1958) would be able to learn it perfectly.

However, the model I describe is not just one static mapping of locations, its aim is to describe behaviour and it can easily be interfaced with the larger framework of dynamic neural fields. In contrast to the machine learning approaches, in the presented model, the internal structure of the task is used to solve it in a very simple way that allows fast learning with very few repetitions of the data. Furthermore, the working principle is easily understandable due to clearly labelled modules and sparse activations, which is not the case for trained neural network models.

Insights into neurobiological questions

We do not understand enough about the details of connectivity and the working principle of the song system to create a really plausible model at the level of detail of spikes. So it might be more efficient to first come up with simple models on a higher level of abstraction, instead of simulation spiking models. In the end, however, we can only really say that we have understood a system, when we can describe it in its own language, which in this case might mean in spiking neurons. However, in spite of all the breakthroughs in rate based neural network models, we still are far from understanding the language of spikes.

Nevertheless, this should not discourage us to explore ideas both from a neuromorphic and from a neurobiological point of view. Although it is controversial, the “constrained forward engineering”-approach might actually work in this context. Especially as long as computation power constraints of classical computers limit simulation speed of spiking networks severely.

Outlook

As the model so far does not provide a biologically plausible mechanism for bird song learning, but is straightforward to extend, it might be interesting to implement more advanced ideas of birdsong learning and include functions of the anterior forebrain pathway that are necessary for variability generation and learning. Once a model works, it might be interesting to look at memory consolidation during sleep or evolution of proto syllables. A model could also make testable predictions about gating of certain synapses during song production or listening.

I plan to implement the presented model in neuromorphic hardware. Taking only the sequence learning core of the model and the two dynamic neural fields, a model for imitation learning in a neuromorphically controlled robot is possible.

If I would like to solely rely on on-chip learning, the model could be implemented on the ROLLS chip (Qiao, Mostafa et al. 2015).

As I already used a compatible neuron equation and the Fusi synapse in the simulation, only one mayor additional constraint has to be met for a neuromorphic implementation: The number of neurons has to be 256 if I use one or 512 if I use 2 connected chips. Furthermore, I have to take into account that due to noise and mismatch, some neurons might be unusable and ordinal, memory and chain nodes have to be implemented as small populations of 5-10 neurons. The number of synapses is not a constraint on the ROLLS chip.

With the following equation, I try to estimate the number of neurons that are used by the model under these circumstances:

$$n_{neurons} = 2 * s^2 + (2 + c) * l * n + 4\text{*s} \quad (17)$$

with
l = number of elements in a sequence
s = field size (1d)
n = number of neurons per node
c = length of chain

With small fields of s = 8, only l=2 sequence elements, n=6 neurons per node and a chain of c=10, the implementation would require already at least 304 neurons and it is unclear if only 10 chain nodes are sufficient to generate a long enough sequence. So ideally, 2 chips are required. If there is one field per chip, if we use the first model variant and if the chain neurons are mirrored on both chips, there should be no need for off chip learning and the amount of data exchanged between the chips remains small.

I suggest, that the model could be used to learn a sequence of 2 short sine sweeps. As a spiking input to the chip I could try to use a simulated or ideally a real silicon cochlea (Liu, Van Schaik et al. 2010). As output a simple speaker would do.

It might also be interesting to implement the first fully neuromorphic WTA-based spiking SOM with on-chip learning. However, before trying that, one would have to show using simulations, that the hardware constraints still allow a well-functioning SOM which I could not show so far.

Conclusion

In this thesis I present a songbird-inspired spiking neural network model for sensorimotor sequence learning based on dynamic neural fields implemented as winner-take-all networks.

The core of the model extends dynamic field theory by adding a representation of time that is used to drive peaks through dynamic fields forming sensorimotor trajectories

The model is applied to birdsong learning and I show in simulations that it is able to learn trajectories of peaks in a winner-take all field and that it can map a sensory to a motor trajectory.

I envision an implementation on a neuromorphic chip. As in my simulations, the main constraints are already accounted for, I suggest to implement a closed loop neuromorphic system that is able to learn a short sequence of sine sweeps.

Acknowledgements

I thank

my supervisor Yulia Sandamirskaya for sharing her knowledge and ideas with me and for giving me the great opportunity to do this project,

Giacomo Indiveri for co-supervision of the project,

Anja Zai who shared songbird sound recordings to be used as tutor sequences,

Richard Hahloser, Gagan Narula and Moritz Milde for fruitful discussions and feedback that helped me to improve the project and see things more clearly,

the NCS group and all people at the institute of neuroinformatics, especially all people who shared their knowledge through discussions, talks and lectures and the people who keep the institute running and help out whenever necessary,

the NSC master students for the great time I had during the last years,

my friends and my family for their support.

Literature

Aldridge, J. W., K. C. Berridge and A. R. Rosen (2004). "Basal ganglia neural mechanisms of natural movement sequences." Canadian journal of physiology and pharmacology **82**(8-9): 732-739.

Amador, A., Y. S. Perl, G. B. Mindlin and D. Margoliash (2013). "Elemental gesture dynamics are encoded by song premotor cortical neurons." Nature **495**(7439): 59-64.

Andalman, A. S. and M. S. Fee (2009). "A basal ganglia-forebrain circuit in the songbird biases motor output to avoid vocal errors." Proceedings of the National Academy of Sciences **106**(30): 12518-12523.

Aronov, D., A. S. Andalman and M. S. Fee (2008). "A specialized forebrain circuit for vocal babbling in the juvenile songbird." Science **320**(5876): 630-634.

Ashmore, R. C., J. M. Wild and M. F. Schmidt (2005). "Brainstem and forebrain contributions to the generation of learned motor behaviors for song." Journal of Neuroscience **25**(37): 8543-8554.

Bahdanau, D., K. Cho and Y. Bengio (2014). "Neural machine translation by jointly learning to align and translate." arXiv preprint arXiv:1409.0473.

Brader, J. M., W. Senn and S. Fusi (2007). "Learning real-world stimuli in a neural network with spike-driven synaptic dynamics." Neural computation **19**(11): 2881-2912.

Brainard, M. S. and A. J. Doupe (2000). "Interruption of a basal ganglia–forebrain circuit prevents plasticity of learned vocalizations." Nature **404**(6779): 762-766.

Brea, J., W. Senn and J.-P. Pfister (2013). "Matching recall and storage in sequence learning with spiking neural networks." The Journal of Neuroscience **33**(23): 9565-9575.

Brette, R. and W. Gerstner (2005). "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity." J Neurophysiol **94**(5): 3637-3642.

Chappelier, J. and A. Grumbach (1996). "A Kohonen map for temporal sequences." Proceeding of neural Networks and Their Application, NEURAP'96, IUSPIM: 104-110.

Chappell, G. J. and J. G. Taylor (1993). "The temporal Kohønen map." Neural networks **6**(3): 441-445.

Chicca, E., F. Stefanini, C. Bartolozzi and G. Indiveri (2014). "Neuromorphic electronic circuits for building autonomous cognitive systems." Proceedings of the IEEE **102**(9): 1367-1388.

Crowe, D. A., W. Zarco, R. Bartolo and H. Merchant (2014). "Dynamic representation of the temporal and sequential structure of rhythmic movements in the primate medial premotor cortex." The Journal of Neuroscience **34**(36): 11972-11983.

Dave, A. S., A. C. Yu and D. Margoliash (1998). "Behavioral State Modulation of Auditory Activity in a Vocal Motor System." Science **282**(5397): 2250-2254.

Douglas, R. J. and K. A. Martin (2004). "Neuronal circuits of the neocortex." Annu. Rev. Neurosci. **27**: 419-451.

Doupe, A. J. and P. K. Kuhl (1999). "Birdsong and human speech: common themes and mechanisms." Annual review of neuroscience **22**(1): 567-631.

Doya, K. and T. J. Sejnowski (1995). "A novel reinforcement model of birdsong vocalization learning." Advances in neural information processing systems: 101-108.

Euliano, N. R. and J. C. Principe (1996). Spatio-temporal self-organizing feature maps. Neural Networks, 1996., IEEE International Conference on, IEEE.

Farries, M. A. and D. J. Perkel (2002). "A telencephalic nucleus essential for song learning contains neurons with physiological characteristics of both striatum and globus pallidus." The Journal of neuroscience **22**(9): 3776-3787.

Fee, M. S. and J. H. Goldberg (2011). "A hypothesis for basal ganglia-dependent reinforcement learning in the songbird." Neuroscience **198**: 152-170.

Fiete, I. R., W. Senn, C. Z. Wang and R. H. Hahnloser (2010). "Spike-time-dependent plasticity and heterosynaptic competition organize networks to produce long scale-free sequences of neural activity." Neuron **65**(4): 563-576.

Franz, M. and F. Goller (2002). "Respiratory units of motor production and song imitation in the zebra finch." Journal of neurobiology **51**(2): 129-141.

Gibb, L., T. Q. Gentner and H. D. Abarbanel (2009). "Inhibition and recurrent excitation in a computational model of sparse bursting in song nucleus HVC." Journal of neurophysiology **102**(3): 1748-1762.

Glaze, C. M. and T. W. Troyer (2006). "Temporal structure in zebra finch song: implications for motor coding." J Neurosci **26**(3): 991-1005.

Graves, A., A.-r. Mohamed and G. Hinton (2013). Speech recognition with deep recurrent neural networks. Acoustics, speech and signal processing (icassp), 2013 ieee international conference on, IEEE.

Hahnloser, R. H., A. A. Kozhevnikov and M. S. Fee (2002). "An ultra-sparse code underliesthe generation of neural sequences in a songbird." Nature **419**(6902): 65-70.

Hanuschkin, A., S. Ganguli and R. H. Hahnloser (2013). "A Hebbian learning rule gives rise to mirror neurons and links them to control theoretic inverse models." Front Neural Circuits **7**: 106.

Harvey, C. D., P. Coen and D. W. Tank (2012). "Choice-specific sequences in parietal cortex during a virtual-navigation decision task." Nature **484**(7392): 62-68.

Indiveri, G., E. Chicca and R. J. Douglas (2009). "Artificial cognitive systems: From VLSI networks of spiking neurons to neuromorphic cognition." Cognitive Computation **1**(2): 119-127.

Indiveri, G., R. Murer and J. Kramer (2001). "Active vision using an analog VLSI model of selective attention." IEEE Transactions on circuits and systems II: Analog and Digital Signal processing **48**(5): 492-500.

James, D. L. and R. Miikkulainen (1995). "SARDNET: A self-organizing feature map for sequences." Advances in neural information processing systems **7**: 577.

Jin, D. Z., F. M. Ramazanoğlu and H. S. Seung (2007). "Intrinsic bursting enhances the robustness of a neural network model of sequence generation by avian brain area HVC." Journal of computational neuroscience **23**(3): 283.

Kohonen, T. (1982). "Self-organized formation of topologically correct feature maps." Biological cybernetics **43**(1): 59-69.

Leonardo, A. and M. S. Fee (2005). "Ensemble coding of vocal control in birdsong." J Neurosci **25**(3): 652-661.

Li, M. and H. Greenside (2006). "Stable propagation of a burst through a one-dimensional homogeneous excitatory chain model of songbird nucleus HVC." Physical Review E **74**(1): 011918.

Liu, S.-C., A. Van Schaik, B. A. Mincti and T. Delbruck (2010). Event-based 64-channel binaural silicon cochlea with Q enhancement mechanisms. Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on, IEEE.

Livi, P. and G. Indiveri (2009). A current-mode conductance-based silicon neuron for address-event neuromorphic systems. 2009 IEEE international symposium on circuits and systems, IEEE.

Long, M. A. and M. S. Fee (2008). "Using temperature to analyse temporal dynamics in the songbird motor pathway." Nature **456**(7219): 189-194.

Long, M. A., D. Z. Jin and M. S. Fee (2010). "Support for a synaptic chain model of neuronal sequence generation." Nature **468**(7322): 394-399.

MacDonald, C. J., K. Q. Lepage, U. T. Eden and H. Eichenbaum (2011). "Hippocampal “time cells” bridge the gap in memory for discontiguous events." Neuron **71**(4): 737-749.

Malnic, B., J. Hirono, T. Sato and L. B. Buck (1999). "Combinatorial receptor codes for odors." Cell **96**(5): 713-723.

Markowitz, J. E., W. A. Liberti III, G. Guitchounts, T. Velho, C. Lois and T. J. Gardner (2015). "Mesoscopic patterns of neural activity support songbird cortical sequences." PLoS Biol **13**(6): e1002158.

Mehaffey, W. H. and A. J. Doupe (2015). "Naturalistic stimulation drives opposing heterosynaptic plasticity at two inputs to songbird cortex." Nature neuroscience **18**(9): 1272-1280.

Mooney, R. (2009). "Neural mechanisms for learned birdsong." Learning & Memory **16**(11): 655-669.

Nowotny, T., M. I. Rabinovich and H. D. Abarbanel (2003). "Spatial representation of temporal information through spike-timing-dependent plasticity." Phys Rev E Stat Nonlin Soft Matter Phys **68**(1 Pt 1): 011908.

Pastalkova, E., V. Itskov, A. Amarasingham and G. Buzsáki (2008). "Internally generated cell assembly sequences in the rat hippocampus." Science **321**(5894): 1322-1327.

Perl, Y. S., E. M. Arneodo, A. Amador, F. Goller and G. B. Mindlin (2011). "Reconstruction of physiological instructions from Zebra finch song." Physical Review E **84**(5): 051909.

Prather, J. F., S. Nowicki, R. C. Anderson, S. Peters and R. Mooney (2009). "Neural correlates of categorical perception in learned vocal communication." Nature neuroscience **12**(2): 221-228.

Prather, J. F., S. Peters, S. Nowicki and R. Mooney (2008). "Precise auditory–vocal mirroring in neurons for learned vocal communication." Nature **451**(7176): 305-310.

Qiao, N., H. Mostafa, F. Corradi, M. Osswald, F. Stefanini, D. Sumislawska and G. Indiveri (2015). "A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128K synapses." Front Neurosci **9**: 141.

Richter, M., Y. Sandamirskaya and G. Schöner (2012). A robotic architecture for action selection and behavioral organization inspired by human cognition. Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, IEEE.

Riesenhuber, M. and T. Poggio (1999). "Hierarchical models of object recognition in cortex." Nature neuroscience **2**(11): 1019-1025.

Rizzolatti, G. and L. Craighero (2004). "The mirror-neuron system." Annu. Rev. Neurosci. **27**: 169-192.

Rosenblatt, F. (1958). "The perceptron: A probabilistic model for information storage and organization in the brain." Psychological review **65**(6): 386.

Roth, A. and M. C. van Rossum (2009). "Modeling synapses." Computational modeling methods for neuroscientists **6**: 139-160.

Rudolph, C., T. Storck and Y. Sandamirskaya (2015). Learning to reach after learning to look: A study of autonomy in learning sensorimotor transformations. Neural Networks (IJCNN), 2015 International Joint Conference on, IEEE.

Ruf, B. and M. Schmitt (1997). Unsupervised learning in networks of spiking neurons using temporal coding. International Conference on Artificial Neural Networks, Springer.

Rumbell, T., S. L. Denham and T. Wennekers (2014). "A spiking self-organizing map combining STDP, oscillations, and continuous learning." IEEE Trans Neural Netw Learn Syst **25**(5): 894-907.

Sandamirskaya, Y. (2013). "Dynamic neural fields as a step toward cognitive neuromorphic architectures." Front Neurosci **7**: 276.

Sandamirskaya, Y., Kreiser, R. (in preparation). "Sequence learning in neuromorphic hardware."

Sandamirskaya, Y. and G. Schöner (2010). "An embodied account of serial order: how instabilities drive sequence generation." Neural Netw **23**(10): 1164-1179.

Scharff, C. and F. Nottebohm (1991). "A comparative study of the behavioral deficits following lesions of various parts of the zebra finch song system: implications for vocal learning." The Journal of neuroscience **11**(9): 2896-2913.

Schmidt, M. F. (2003). "Pattern of interhemispheric synchronization in HVc during singing correlates with key transitions in the song pattern." Journal of neurophysiology **90**(6): 3931-3949.

Schöner, G. and J. Spencer (2015). Dynamic thinking: A primer on dynamic field theory, Oxford University Press.

Sutskever, I., O. Vinyals and Q. V. Le (2014). Sequence to sequence learning with neural networks. Advances in neural information processing systems.

Titze, I. R. (1988). "The physics of small-amplitude oscillation of the vocal folds." The Journal of the Acoustical Society of America **83**(4): 1536-1552.

Tumer, E. C. and M. S. Brainard (2007). "Performance variability enables adaptive plasticity of 'crystallized' adult birdsong." Nature **450**(7173): 1240-1244.

Varstal, M., J. D. R. Millán and J. Heikkonen (1997). A recurrent self-organizing map for temporal sequence processing. International Conference on Artificial Neural Networks, Springer.

Vinyals, O., A. Toshev, S. Bengio and D. Erhan (2015). Show and tell: A neural image caption generator. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Vu, E. T., M. E. Mazurek and Y.-C. Kuo (1994). "Identification of a forebrain motor programming network for the learned song of zebra finches." The Journal of neuroscience **14**(11): 6924-6934.

Wild, J. M. (1993). "Descending projections of the songbird nucleus robustus archistriatalis." Journal of Comparative Neurology **338**(2): 225-241.

Zen, H. and H. Sak (2015). Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on, IEEE.