Neurobiological models of sentence processing

Marvin Uhlmann

The educational component of the doctoral training was provided by the International Max Planck Research School (IMPRS) for Language Sciences. The graduate school is a joint initiative between the Max Planck Institute for Psycholinguistics and two partner institutes at Radboud University – the Centre for Language Studies, and the Donders Institute for Brain, Cognition and Behaviour. The IMPRS curriculum, which is funded by the Max Planck Society for the Advancement of Science, ensures that each member receives interdisciplinary training in the language sciences and develops a well-rounded skill set in preparation for fulfilling careers in academia and beyond.

More information can be found at www.mpi.nl/imprs

ISBN: 978-94-92910-17-2

Cover design: Maria Otworowska & Marvin Uhlmann

Printed by: Ipskamp printing

Layout: This thesis was typeset with LaTeX 2_ε. It is based on the Clean Thesis style, adapted by Maria Otworowska.

The research reported in this thesis was conducted at the Max Planck Institute for Psycholinguistics, in Nijmegen, the Netherlands.

Neurobiological models of sentence processing

Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de rector magnificus prof. dr. J.H.J.M. van Krieken,
volgens besluit van het college van decanen
in het openbaar te verdedigen op

dinsdag 29 september 2020
om 10.30 uur precies

door

Marvin Uhlmann

geboren op 21 oktober 1988
te Lutherstadt Wittenberg (Duitsland)

Promotor
Prof. dr. P. Hagoort

Copromotoren
Dr. K.M. Petersson (MPI Nijmegen)
Dr. H. Fitz (MPI Nijmegen)

Manuscriptcommissie
Prof. dr. M.A.J. van Gerven
Prof. dr. T. Celikel
Dr. A.E. Martin (MPI Nijmegen)

Neurobiological models of sentence processing

Doctoral Thesis

to obtain the degree of doctor
from Radboud University Nijmegen
on the authority of the Rector Magnificus prof. dr. J.H.J.M. van Krieken,
according to the decision of the Council of Deans to be defended in public on

Tuesday, September 29, 2020
at 10.30 hours

by

Marvin Uhlmann

Born on October 21, 1988
in Lutherstadt Wittenberg (Germany)

Supervisor
Prof. dr. P. Hagoort

Co-supervisors
Dr. K.M. Petersson (MPI Nijmegen)
Dr. H. Fitz (MPI Nijmegen)

Doctoral Thesis Committee
Prof. dr. M.A.J. van Gerven
Prof. dr. T. Celikel
Dr. A.E. Martin (MPI Nijmegen)

Contents

1

Introduction

An important aim of psycholinguistics is to understand how different aspects of language are implemented in the brain. These processes, including comprehension, production, and acquisition, are assumed to be computational. For each computation there can exist many different realizing algorithms and different ways to implement the algorithm in different computational systems, including the brain (Marr, 1982; Fodor, 1974). Cognitive models in psycholinguistics are based on empirical data from behavioral and neuroimaging experiments and use this data to derive an algorithmic description of language processing (Altmann, 1995). However, because they typically do not take into account the behavior and limitations of biological circuits, they have little explanatory power when it comes to understanding how such an algorithm may be implemented in the brain on the level of neurons, synapses and networks. In this dissertation, I introduce the notion of *neurobiological models* that take into account the biophysical constraints imposed by the neuronal infrastructure and through this connect cognitive data with the properties of neurons and networks.

1.1 Neurobiological models

One of the aims of a neurobiological model is to reflect the constraints imposed by known neurophysiology and neurobiological infrastructure. Based on these constraints, the model processes information which can then be related to aspects of cognition, behavioral or cognitive neuroscience data. Such a model is built using experimentally attained insights about the structure and function of neurons, and the connectivity of neural networks in the brain. Thus, the aim is to capture relevant properties of the biological system that implements the information processing. This allows for the investigation of constraints on processing imposed by the neurobiology of the underlying system and provides the basis for a mechanistic understanding of language processing.

Cognitive and neurobiological models can be of different levels of complexity. For cognitive models, simplicity is often preferred if this does not sacrifice coverage and precision in the phenomena the model is attempting to capture. This is typically not the case for neurobiological models which aim to reflect properties of the underlying biological system. Because the biological system consists of large networks of neurons with interconnecting synapses, supporting a variety of biophysical processes, neurobiological models can have a large number of independent components and parameters that determine how they respond to input. However, the characteristics of the biological system impose constraints on those parameters and therefore determine the model properties. This means that the values of model parameters are provided by independent experimental data from fundamental neuroscience. Implementing such neurobiological models implies, among other things, that the model parameters also have physical units of measurement that need to fall within physiological bounds. Another implication is that these models operate physical time that can be directly related to temporal aspects of behavioral and neuroimaging experiments.

However, the goal of neurobiological modeling is not to reproduce the entire complexity of the biological system, but rather to understand the system and its functionally relevant aspects in models of reduced complexity. Hence, the goal of such models is to reflect relevant characteristics of the neurobiological system, while at the same time relying on some level of abstraction or reduction of the full details of the neural system, in particular, at the neuronal level. These reductions are derived from experimental data and attempt to capture critical properties that have significant effects on the behavior of neurons and networks in which they are integrated. Neurobiological models can then be used to investigate the causal relationship between biophysical properties and the processing characteristics of neural networks in an attempt to relate these to cognition. The explicit model descriptions (typically a large set of coupled differential equations) are implemented as computer simulations which provides complete control and manipulation opportunities of the various components and parameters of the model. Models with different implementational features can be compared to establish relationships between these aspects of a model and its processing characteristics. Since the models reflect properties of the actual neurobiology, these causal relationships are not only relevant within the models but also for the functional interpretation of the behavior of actual biological subsystems of the brain. Ultimately, neurobiological modeling aims to bridge the description

Figure 1.1.: Image based on silver staining from Ramón y Cajal (around 1905). The capital letters indicate different morphological neuron classes within the pyramidal neurons (A to G) and other morphological neuron classes (H to M). Small *a* indicate the axon protruding out from each cell. The cortical layer structure is indicated by numbers on the left.

of brain operations across levels of explanation in order to provide a mechanistic understanding of higher-level cognitive functions, including language processing.

1.2 Intersecting disciplines

The work presented in this dissertation applies methods from computational neuroscience and neurophysiology to address questions in psycholinguistics and cognitive neuroscience. To give a comprehensive introduction to these fields goes beyond the scope of this chapter. However, I will introduce the findings and concepts that this dissertation is based on and that are used later to argue for or against particular design choices in modeling (chapter 2).

1.2.1 Neurophysiology

The brain is a large network of interconnected neurons. To understand how information is processed in this system, it is important to understand the properties of neurons as developed experimentally in the field of neurophysiology.

Single neurons

Biological neurons are complex molecular machines and their experimental study was initiated in the late 19th century by Ramón y Cajal, see Figure 1.1. Based on his morphological characterizations, he suggested that the neurons in the brain are fundamental computational units that communicate with each other. Neurons typically consist of three basic structural components: (i) a dendritic tree, receiving and accumulating electric signals, (ii) a soma or cell body with the nucleus and the axon hillock where action potentials are generated, and (iii) an axon that transmits the action potentials to synapses, and through chemical transmission via synaptic release of neurotransmitters, relay signals to other, post-synaptic neurons (Figure 1.2A).

Action potentials are short-lived electrical pulses that are generated and transmitted through a chain reaction of opening of voltage-gated ion channels in the cell membrane. Action potentials are triggered in the axon hillock when a membrane threshold is passed. The chain reaction is not locally contained but triggers ion channel openings in neighboring sections of the axon membrane, leading to an electric pulse traveling down the axon of the neuron. Action potentials are often called *spikes* because they exhibit a rapid ($\sim$1ms) local increase and subsequent decrease of the membrane potential (as in Figure 1.2B). At the end of axons, the pre-synaptic nerve

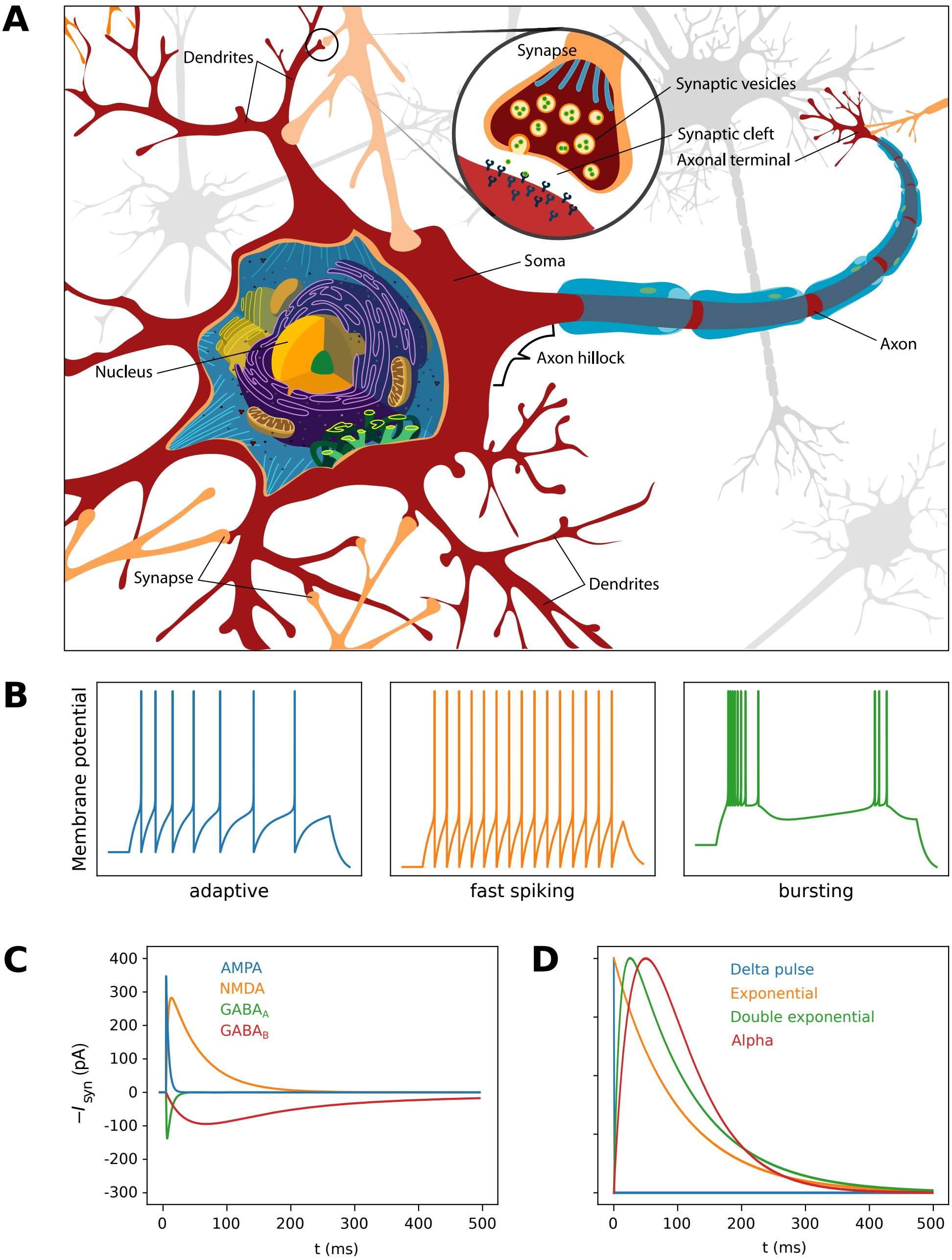


Figure 1.2.: **A:** Neurophysiological details of a typical neuron. **B:** Spiking response of three different electrophysiological neuron classes to a constant input current. Fast spiking neurons are a subclass of the non-adaptive neurons (J. R. Gibson et al., 1999). **C:** Post-synaptic current models for the four major synapse classes, adapted from Gerstner et al. (2014). **D:** Shapes of different synapse models described in the main text. Note that the Dirac delta pulse synapse is zero everywhere except at time 0.

terminals, there are typically several thousand synapses interconnecting hundreds or thousands of different post-synaptic neurons. When these synapses are activated through a spike, they trigger an electrical current in each post-synaptic neuron by changing the conductance of ligand-gated ion channels embedded in the cellular membrane of the post-synaptic neuron. These synaptic currents can be long-lived (up to $\sim$1 second, Gerstner et al., 2014) and propagate down the dendritic tree towards the soma. In this process, currents from synapses in the dendritic tree are integrated. Integration is not necessarily a simple, linear addition in time but is typically branch and distance-dependent and can be non-linear, in particular if there are active, voltage-dependent ion channels in the dendritic membrane (Koch, 1999). For example, the stimulation of two adjacent synapses on the same dendritic branch has a different effect on the neuronal spiking compared to the stimulation of two synapses on different branches. This effect can be both enhancing and suppressing (Bono & Clopath, 2017). While synaptic currents can have a long duration, the communication delay from one cortical neuron to the next is short and typically take less than 10 ms (Markram et al., 2015; Izhikevich & Edelman, 2008). This includes the time for the action potential to travel down the axon of the pre-synaptic neuron and the opening of ligand-gated ion channels in the post-synaptic neuron.

The spiking behavior of neurons can be studied with patch clamp experiments, where an electric current is directly applied to a neuron in order to generate spikes. It was discovered that neurons do not simply spike whenever a fixed amount of current has accumulated (Koch, 1999). Instead, neurons exhibit a range of spiking behaviors (Connors & Gutnick, 1990; J. R. Gibson et al., 1999). The specific behavior defines different electrophysiological neuron classes (Izhikevich & Edelman, 2008; Markram et al., 2015). The three most prominent classes are adaptive, non-adaptive and bursting neurons. Figure 1.2B shows an example of the spike response to a constant input current for each of the three classes. Biological neurons often exhibit history-dependence in that the spike response to input depends on past input, a form of intrinsic plasticity (Titley et al., 2017). One process realizing intrinsic plasticity is spike rate adaptation. If a neuron with spike rate adaptation, such as an *adaptive neuron* (see Figure 1.2B), is stimulated with a constant current, then, if the current is strong enough, the neuron will initially fire rapidly but, despite uninterrupted stimulation, after a few spikes the firing activity will slow down (Connors & Gutnick, 1990). In addition to intrinsic plasticity, there is a range of various other plasticity mechanisms in neurons.

Besides the classification of neurons by electrophysiological properties, there are other ways to distinguish different neuron classes, such as *morphology*. Neuronal morphology describes the way dendrites and axons branch out from the soma. An early example of a morphological study is shown in Figure 1.1. However, the different classifications can not be directly mapped to each other in that members of one electrophysiological neuron class can belong to several different morphological classes and vice versa (Markram et al., 2015). But a reliable functional classification can be made in terms of excitatory and inhibitory neurons. Whether an axonal signal is exciting or inhibiting a post-synaptic neuron depends on the type of synapse and, more precisely, the chemical transmitters released by the pre-synaptic neuron and the type of ligand-gated ion channels in the post-synaptic membrane. It turns out that a given neuron only excites or inhibits its post-synaptic neurons, but never a combination of both. This distinction is also reflected in electrophysiology and the morphological structure and is referred to as Dale's principle (Strata & Harvey, 1999). Excitatory neurons are overall more common ($\sim$80 percent) in cortex and they show less morphological and electrophysiological diversity. The majority are pyramidal neurons which typically exhibit adaptive spiking behavior (Markram et al., 2015). Inhibitory neurons are less common ($\sim$20 percent) but are more diverse in terms of both morphology and electrophysiological behavior. Many are non-adaptive and exhibit regular spiking behavior (Markram et al., 2015).

Synapses

A synapse translates action potentials from the pre-synaptic neuron into dendritic currents in the post-synaptic neuron. It consists of a synaptic bulb at the pre-synaptic neuron which releases neurotransmitters via a narrow synaptic cleft to the post-synaptic neuron (Figure 1.2A). Synapses can either cause an excitatory or inhibitory post-synaptic current, both of which can be expressed through several different synapse types (Markram et al., 2004). However, there are two major classes of excitatory synapses, AMPA and NMDA synapses, and two major classes of inhibitory ones, GABA_A and GABA_B synapses (Gerstner et al., 2014) and these generate different post-synaptic currents (Figure 1.2C). The functional difference between these groups is in the temporal dynamics of the post-synaptic current they generate. AMPA and GABA_A synapses generate comparatively brief currents with a duration of approximately 5 ms while NMDA and GABA_B synapses generate more temporally extended currents that can last up to 1000 ms (Destexhe et al., 1994; Destexhe & Paré, 1999; Gerstner et al., 2014). The biochemical interactions in synapses are complex and synapses show a variety of adaptive behaviors (Li et al., 2010). I will discuss models for synaptic dynamics below but beyond this, there are also molecular interactions at play that are driven by protein synthesis, transport, and state changes by covalent modification (for instance kinase-catalyzed phosphorylation) and binding of various chemical factors (K. C. Martin & Kosik, 2002; Fallon & Taylor, 2013).

Networks of neurons

While a neuron in itself is a complex entity, the network of neurons in the brain adds another layer of complexity. On the largest scale, the neurons of the cerebral cortex only inhabit a small sheet of tissue close to the heavily folded surface of the brain (van Essen et al., 1998). The interior of the brain contains subcortical structures (e.g., the basal ganglia, thalamus and cerebellum) as well as neuronal fiber tracts connecting parts of the cortex with each other, and with subcortical structures. Based on cytoarchitectonic differences (Brodmann, 1909; Strotzer, 2009) and neuroimaging (van Essen et al., 1998; Geyer et al., 2011), the cortex has been parcellated into areas that form connected but functionally distinct modules. The connectivity between these areas, each containing several hundreds of millions of neurons, is broadly consistent across humans (Sporns et al., 2005; van den Heuvel & Sporns, 2011). The neurons in cortex are not uniformly distributed but form a layered structure. Figure 1.1 shows a part of this layered structure. For example, there are many inhibitory neurons close to the cortical surface, while further away from the surface cortex is mainly populated by large excitatory pyramidal neurons (Markram et al., 2015). The connectivity between neurons of different layers seems to induce a functional processing order. Layer 4 serves as input layer from subcortical structures. Information is then forwarded and processed in the top layers (layers 1-3) and layer 5/6 generates output to other cortical and subcortical brain regions. Communication between neighboring cortical regions happens through neurons close to the cortical surface in layer 1 (Feldmeyer & Sakmann, 2000; Izhikevich & Edelman, 2008).

On the smallest scale, this leaves the connectivity of neurons within one layer. There is ongoing, active research that characterizes reoccurring motifs and connectivity patterns that deviate from random connectivity (Sporns & Kötter, 2004; Song et al., 2005; Perin et al., 2011). However, the neuronal connectivity at this scale seems to be best described by connection probabilities between neurons that depend on the classes of the sending and the receiving neuron, their synaptic types, coupled with morphological properties of the neuron classes involved. These probabilities are also mediated by the distance between the two neuronal somata (Reimann et al., 2015; Markram et al., 2015).

1.2.2 Computational Neuroscience

Computational neuroscience research conducts *in silico* experiments by simulating neurons or neuronal populations. These experiments are often limited by the computational power that is available. This means that, typically, single neurons or small neuron populations are simulated with high physiological detail while simulations of larger neuron populations rely on simpler neuron models. Notable exceptions are various large scale projects including, for example, the human brain projects (Markram et al., 2015), that expand both the level of detail and size of the simulated networks.

One feature that is typically omitted in simple neuron models is the spatial extent of a neuron – the dendritic tree and the physical distance between the synapses and the soma of the neuron. Instead, neurons are simulated as point neurons, that is, they have a single compartment that receives synaptic input, integrates the input and generates spikes to be transmitted to post-synaptic neurons. Such neuron models can be simulated using only a few coupled differential equations which contrasts with more complex neuron models that can require tens or hundreds of differential equations. An example of a simple point neuron model is the leaky integrate-and-fire (LIF) neuron. An LIF neuron integrates input additively and has a leak process that causes the membrane potential u to revert to a resting state after a time determined by the neuron's membrane time constant τ_m. The behavior of an input-free LIF neuron is described by $\tau_m \frac{du}{dt} = -u$. When the membrane potential reaches a firing threshold, a spike is emitted and the membrane potential is reset to a reset potential (see Dayan & Abbott, 2001). This neuronal model does not have long-term dynamics, that is, when the LIF neuron emits a spike, information about previous input is forgotten (Gerstner et al., 2014). Neuronal models that can maintain information beyond spike emission, as is the case for a biological neuron, need at least two dynamical variables. Two-dimensional neuron models use similar spiking and integration mechanisms as the LIF neuron but have additional dynamical variables that maintain information beyond spike emission. This can be, for example, an adaptive current or an adaptive threshold. Examples of those neurons models are the FitzHugh–Nagumo model (FitzHugh, 1961), the Izhikevich model (Izhikevich, 2003) and the AdEx model (Brette & Gerstner, 2005).

These phenomenological models approximate the spiking behavior of neurons instead of the underlying biophysical processes. However, it is possible to model the processes that give rise to a spiking behavior directly, as was done in the Hodgkin-Huxley model (Hodgkin & Huxley, 1952). It uses a set of differential equations that describe ion channel interactions which can model spike generation without an additional spike mechanism. However, through theoretical analysis, phenomenological two-dimensional neuron models, such as the AdEx neuron, can be derived from the biophysical Hodgkin-Huxley model using separation of time scales (Brette & Gerstner, 2005; Gerstner et al., 2014; Brette, 2015b). This means these phenomenological models capture properties of the neuronal behavior that can be directly derived from the Hodgkin-Huxley model. Furthermore, despite the added biophysical details in the Hodgkin-Huxley model, the AdEx model reflects the neuronal spiking and adaptation behavior of real cortical neurons most accurately and predicts the spiking response to input the best (Brette, 2015b; Jolivet et al., 2008).

Synapse models

Within synapses, there are a host of biochemical interactions that control the conductivity of a synapse on a range of different time scales. In computational models, these processes are accounted for by the synapse model. Due to the complexity of synaptic dynamics, detailed synapse models can be computationally expensive and there are a number of models with different levels of detail.

When a spike excites a synapse, it generates a conductance response and a corresponding post-synaptic current, the amplitude of which depends on the strength of the synapse, the synaptic weight. The post-synaptic current can either be modeled directly using a current-based model or indirectly using a conductance-based model. The latter is biophysically more detailed as it captures the change in synaptic conductance through ligand-gated ion channel openings rather than the phenomenological effect of the resulting current. It also captures the fact that post-synaptic currents are dependent on the membrane voltage by including different reversal potentials for inhibitory and excitatory synapses. In current-based models, the temporal evolution of the post-synaptic currents can be modeled with different levels of detail (see Figure 1.2D). The simplest model is a delta pulse current that, when a spike is received, instantaneously increases the membrane potential of the post-synaptic neuron by a value depending on the synaptic strength. This model does not generate temporally extended currents, that is, currents only influence the post-synaptic neuron at the moment of the spike arrival. A more complex model is the exponential post-synaptic current, with an initial current depending on the synaptic strength which then exponentially decays over time. It is essentially a Dirac delta convolved with an exponentially decaying synaptic kernel such that the post-synaptic current is temporally extended depending on the decay time constant of the exponential kernel. Further refinements of the shape of the post-synaptic current are the alpha synapse and the double-exponential synapse. In both models, the synaptic current decays exponentially. However, currents do not reach their maximum amplitude at the time of the spike arrival but after a short rise-time. This allows spikes to have a delayed effect on post-synaptic neurons (Dayan & Abbott, 2001).

Synapses also exhibit adaptive properties that change the synaptic weights depending on past input. These mechanisms include short-term plasticity (STP) (Markram et al., 1998) and spike-time dependent plasticity (STDP) (Markram et al., 1997; Kempter et al., 1999) which can work in parallel with longer-term consolidation mechanisms (Frey & Morris, 1997; Zenke et al., 2015). STDP is an example of Hebbian plasticity that implements the principle "fire together, wire together". When Hebbian plasticity is added to a network, synapses and thus network activity have a tendency to become dynamically unstable in that Hebbian plasticity tends to increase strong synapses and depress weak ones. This drives synapses to bottom out and can lead to a bimodal distribution of synaptic strengths (Morrison et al., 2008; Carlson et al., 2013 and already Rochester et al., 1956). This type of bimodal distribution is not what has been observed in brain networks (Turrigiano et al., 1998; Song et al., 2005; Bartol et al., 2015) and there are several proposals to address this problem (Morrison et al., 2008; Kumar et al., 2010; Zenke et al., 2015) but so far none of them has resolved the issue of Hebbian instability conclusively.

Network dynamics

Another feature observed in mammalian brains is that large-scale neuronal spike patterns have statistical properties similar to spikes generated by a (random) Poisson process (Shadlen & Newsome, 1998) where the inter-spike-intervals will approximately follow an exponential distribution. This spike pattern activity in a network is referred to as asynchronous irregular behavior and provides a large dynamic range that can be used to represent information (Duarte & Morrison, 2014). To achieve this state, a local balance of inhibitory and excitatory synaptic input to neurons is crucial (van Vreeswijk & Sompolinsky, 1996; Tsodyks & Sejnowski, 1995). In a simple implementation, this can be approximated globally by balancing the total synaptic strength of inhibition and excitation in a network.

1.2.3 Models in psycholinguistics

In psycholinguistics, models have been used to describe different aspects of how humans comprehend, produce and acquire language. Models of language comprehension describe the cognitive processes at different levels of granularity. Some models address word identification and lexical access (Cohort model, Marslen-Wilson & Tyler, 1980; TRACE model, McClelland & Elman, 1986) while others investigate sentence-level interpretation assuming that words have already been identified correctly (Unification model, Vosse & Kempen, 2000; Sentence Gestalt model, St. John & McClelland, 1990). Different models of language processing can rely on different sources of experimental data. While all, in one way or another, use data from behavioral experiments of humans processing words or sentences, some conceptual models also relate to neuroimaging data of language processing (MUC model, Hagoort, 2005; Dual Stream model, Hickok & Poeppel, 2007). Below, I discuss previous language models that are of particular relevance for this dissertation.

MUC model

This dissertation is conceptually operating within the Memory, Unification, Control (MUC) model (Hagoort, 2005). The MUC model bridges from a psycholinguistically motivated analysis of language processing to evidence from neuroimaging experiments. It postulates three functional subcomponents of language processing. The memory module, or mental lexicon, is a long-term storage of words and their associated phonological, semantic and syntactic features which are retrieved during comprehension. Neurobiologically, it is located in regions of the temporal and the parietal cortex (Angular gyrus). It also stores information such as word forms, morphological information and syntactic templates representing, for instance, nouns, verbs, and adjectives (Hagoort, 2003, 2005, 2013). The unification module integrates information retrieved from the mental lexicon and generates larger structures representing the syntax and semantics of multi-word utterances. It is located in frontal areas around Broca's region and the distribution of activation during unification is dependent on the type of information unification is operating on. Phonological, syntactic, and semantic unification happens in a gradient structure including Brodmann areas BA 44, 45, and 47 (Hagoort, 2013). Finally, the control module exerts higher-level cognitive control. It is involved in, for example, target language selection, turn-taking, joint action, and generally larger-scale discourse contexts (Hagoort, 2013; Meyer et al., 2007). It is located in the dorsolateral prefrontal cortex and mid-line frontal structures (Hagoort, 2013).

Unification is based on a computational model of parsing where each word in the mental lexicon is associated with a set of structural frames (Vosse & Kempen, 2000). These frames can be represented as treelets defining suitable linguistic environments for each word. They consist of a root node, functional nodes (such as subject, direct object, or determiner) and leaf nodes. Each treelet has a head, which is the lexical item it corresponds to, marking its position within the treelet. When perceiving a multi-word utterance, the unification operation binds/unifies the root node of a treelet retrieved from the mental lexicon to an open leaf node of an already constructed, partial parse of previously received treelets with the identical label while ensuring that number, gender, case, inflection etc. agree. This leads to a tree structure representing sentence-level structure and meaning (Hagoort, 2003). Unification operations resolve syntactic as well as semantic constraints (Jackendoff, 2002, 2007; Hagoort, 2013).

Connectionist models

Connectionist models are computational models in psycholinguistics that use artificial neural networks which process information through spreading activation rather than the explicit manipulation of variables and symbols (Frank et al., 2019). An artificial neural network typically consists of a set of simple processing units that are interconnected. An external stimulus can induce activation in the nodes that propagates through the network via synaptic connections. The response of such a network is determined by the activation of a set of output nodes. Since their introduction (Rumelhart et al., 1986; Dell, 1986; Elman, 1990; St. John & McClelland, 1990), connectionist models have been very influential in the field of psycholinguistic modeling (Seidenberg, 1993; Christiansen & Chater, 2001; Frank et al., 2019). What distinguishes them from symbolic models is that they were inspired by structural aspects of biological neural networks. They were built on the observation that the brain is not composed of a single CPU, a dedicated addressable memory, and other specialized components but rather consists of a large number of similar, interconnected processing units. In combination, these can perform complex operations in a distributed fashion, which has some similarity to the way brain networks process information. In this sense, connectionist models have some common roots and share some characteristics with neurobiological models. However, connectionist models do not go beyond implementing decentralized processing and do not model neuronal activity and synaptic dynamics as biophysical processes in real physical time. Nevertheless, by imitating structural characteristics of the brain, connectionist models were useful to model behavioral data in psycholinguistics, including aspects of language acquisition, adult processing, and impairment (Plunkett & Marchman, 1991; Christiansen et al., 1998; Joanisse & Seidenberg, 1999; Chang et al., 2006; Chang, 2009; Fitz & Chang, 2008, 2017).

1.3 Overview of the dissertation

The goal of this dissertation is to build and evaluate a neurobiological model of language processing. In chapter 2, I describe one instance of such a model, the NBL model. Based on the background provided above, I will motivate the model choices, pertaining to the neuron and synapse model, as well as other components such as the encoder and readout. Chapter 2 also contains the general methods describing the underlying mathematics for the simulations, and the English-like language that is used to stimulate the model. The remainder of the dissertation can be split into two parts. Chapters 3 and 4 are dedicated to investigating the NBL model and to understanding how model parameters affect memory and processing, while chapters 5 and 6 evaluate the model in relation to human processing characteristics. Figure 1.3 shows the NBL model and which aspects of the model are investigated in each chapter.

In chapter 3, I compare different ways to encode information into the network and how these different encoding schemes influence information processing. I compare encodings that are based on spike rates, the precise timing of spikes, or a direct current injection. I test if these encodings require a stimulus-specific input projection, where only a dedicated subset of neurons is targeted by each stimulus, or if they can function without it. I focus on processing properties that are important for language comprehension and other cognitive functions: input separation, memory retrieval, and integration. I find that the direct current and temporal encodings perform similar while rate-based encoding shows lower accuracy. In particular, rate encoding does not perform well without a stimulus-specific projection. Then, I test to what extent noise affects processing and find that all three encoding schemes are tolerant to noise with a stimulus-specific projection while without it, temporal encoding performs best under noise.

In chapter 4, I investigate how network features influence the memory characteristics of the NBL model. I manipulate connectivity, neuronal adaptation, and the duration of synaptic currents to determine how they affect processing memory. I

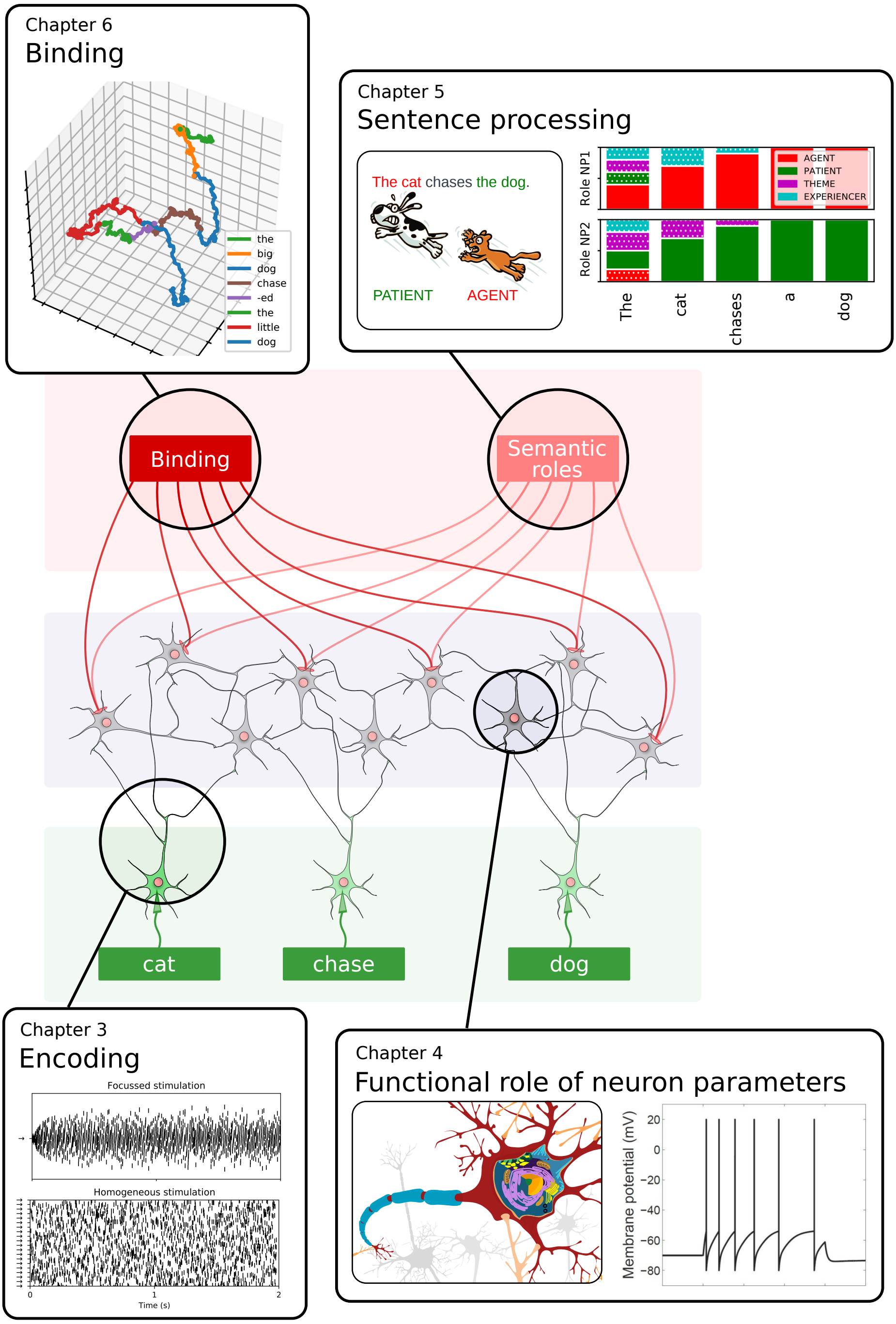


Figure 1.3.: Visual overview of the dissertation. The NBL model is illustrated in the center with its three components encoder (green), network (blue) and readout (red). The surrounding boxes indicate which parts of the model are investigated in the different chapters.

compare recurrent and feed-forward networks with different connection densities and find that processing memory in the NBL model is not supported by recurrent connectivity. Furthermore, I compare networks with and without spike rate adaptation and different synaptic currents. I find that, generally, long time scales in these components enhance memory. Both of these mechanisms do not rely on ongoing spike interaction. That is, they serve as an activity-silent processing memory.

In chapter 5, I use the NBL model with the parameters determined in the previous chapters to processes structured sequences. I investigate if it provides a suitable substrate to accomplish a semantic analysis of sentences. I find that the model readout can be estimated on a small sample of the input language and still correctly generalizes to novel sentences. It integrates new information and retains relevant old information to processes words in the context in which they occur. The model integrates both syntactic and semantic constraints. Furthermore, I show that the NBL model can infer the semantic role of novel words from context.

In chapter 6, I investigate the binding of words to semantic roles using a question-answering protocol. That is, after presenting a sentence, I query the model with a semantic role, such as the agent or the patient of an action, and find that the nouns that correspond to these roles can be decoded from the network activity. Using parallel readouts, I demonstrate that the network maintains different kinds of information concurrently, such as noun and verb identity, but also which adjective is associated with which noun. In addition, the model maintains the event semantics of a sentence. I also find that this information is sufficient to resolve binding relations for nouns that occur repeatedly in a sentence – so-called problem-of-2 sentences. Finally, I explore the state-space dynamics of the NBL model during processing to investigate how it stores binding information.

Chapter 7 discusses how the results of this dissertation exploit and illustrate the benefits of neurobiological models. Furthermore, it outlines possible extensions of this research and how using neurobiological models may facilitate scientific progress and improve our understanding of the language system.

2

The NBL model, a neurobiological model of sentence processing

In this chapter, I describe the NBL model that was used in all simulations in this dissertation. First, I describe the model structure, introduce the main components and argue for the modeling choices I made. Second, I give a description of the implementation details in the general methods section including the underlying mathematical formalism and the model parameters. Third, I describe the English-like language I used to evaluate the model. And finally, I briefly introduce some mathematical terms and concepts that I rely on throughout.

2.1 The NBL model

2.1.1 The neuronal and network model

Neurobiological models and their processing characteristics are important tools in the attempt to understand how various cognitive processes arise from the underlying neural infrastructure. Therefore, I built a model that captures a number of important biological features. The defining feature of a neural network is that it consists of many interconnected processing units that in biology communicate through spikes. Thus, I use networks of models of spiking neurons as a basis for the neurobiological model. The communication between biological neurons is fast but can cause long-lasting effects on the post-synaptic neurons due to temporally extended synaptic currents. I model these interactions with current-based exponential synapses. The currents have instantaneous rise time and are initiated immediately after receiving a spike and then decay exponentially over tens or hundreds of milliseconds (see Figure 1.2D). An important aspect of biological neurons is that they exhibit intrinsic plasticity. Thus, information can be maintained beyond spike generation for a hundred milliseconds and longer. One intrinsic plasticity mechanism is spike rate adaptation, which is well described by the AdEx model (Brette & Gerstner, 2005). The AdEx neuron is a two-dimensional spiking neuron model that models spike rate adaptation through an adaptive current. It reproduces the spiking behavior of neurons more reliably than other more complex models (Brette, 2015b). The equations and parameters defining the neuron and synapse elements are given in section 2.2.1. The functional effects of different parameters values are investigated in chapter 4.

In neurobiology, the most prominent difference between neuron classes is the distinction between excitatory and inhibitory neurons (Strata & Harvey, 1999). In the simulated networks, this distinction is reflected by splitting neurons into two groups that then only generate excitatory or inhibitory synaptic currents in post-synaptic neurons, respectively. In cortex, there are more excitatory than inhibitory neurons. In mammalian cortical layer 2/3, which is relevant for processing (Izhikevich & Edelman, 2008), this ratio is approximately four to one (Markram et al., 2015). I use the same ratio in the simulated network: one fifth of the neurons were inhibitory. To reach a global balance between excitation and inhibition, the synaptic weights of inhibitory neurons were scaled up by five compared to the excitatory synapses. Within cortical layers, connectivity appears random at various spatial scales, with the connection probability between two neurons being influenced by neuron class and distance (Reimann et al., 2015). Due to their small size, the networks modeled here can be compared to a local network within one cortical layer. I therefore use a random connectivity matrix between neurons. Furthermore, since neuron classes are not distinguished beyond excitatory/inhibitory, the connection probability between two neurons is always the same.

These represent minimal neurobiological characteristics of spiking networks and do not include many important aspects of neurobiology that can be simulated with known computational models. More heterogeneous neurons, structured connectivity profiles that differentiate by neuron class, and synaptic dynamics are some of the areas where the neurobiological realism of the model can be improved. However, research on all of these topics and how different aspects of them are interacting is still ongoing. The focus of this dissertation is to establish neurobiological models

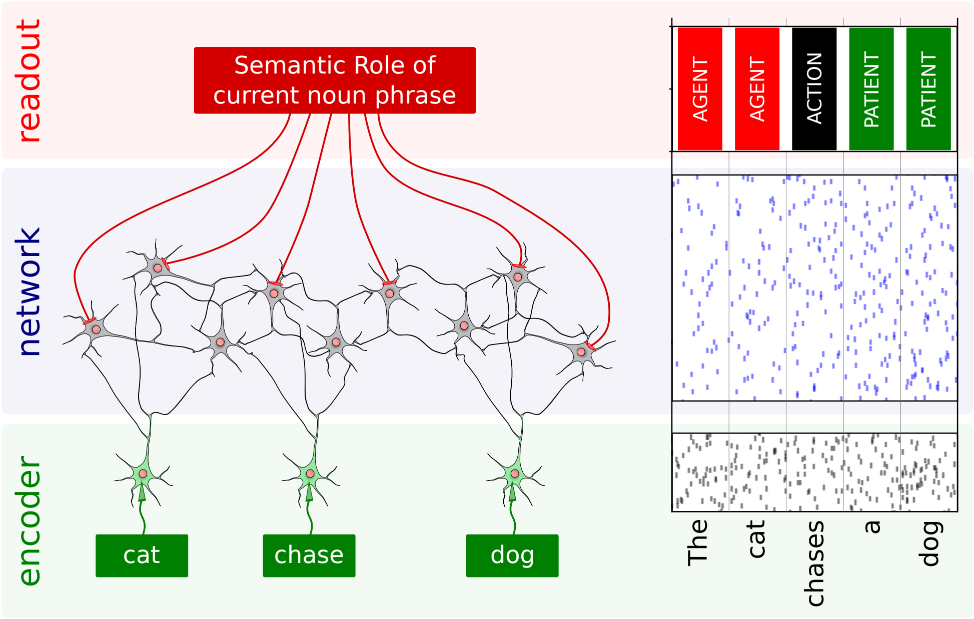


Figure 2.1.: Illustration of model setup. The model components (left) are shown, together with the activity of each component in a simulation (right) during semantic processing (see chapter 4). The encoder maps sentence-like structured sequences into spike patterns that excite the network. The network processes the delivered input and the readout extracts processed information from the network. In the example task, this information is used to assign semantic roles to the current noun phrase. During the presentation of the verb, ACTION is assigned as semantic role.

of language processing and showcase their potential utility. Nevertheless, greater neurobiological realism is an important future research trajectory (see chapter 7).

2.1.2 Model components

To investigate processing in a spiking neural network, there needs to be some means to encode input, some way to measure the result of the processing, and some long-term learning or adaptation in order to accomplish a particular task. Therefore the NBL model contains two components besides the neural network (Figure 2.1): the *encoder*, that stimulates the network in a suitable manner with sequential word input, and the *readout*, that serves as an output device which can be used to identify the solution to a processing task. Neurobiologically, long-term storage and learning are associated with synaptic connections between neurons that dynamically change depending on the input (Koch, 1999; Takeuchi et al., 2014). The network in the NBL model does not have dynamic synapses and can not aggregate information over a whole training set containing tens of thousands of stimuli. Therefore, a readout is calibrated/estimated to solve a specific task by identifying which aspects of the network activity are important in solving this task.

2.1.3 Encoder

During processing, the NBL model should be in a neurobiologically plausible state and it should be stimulated in a neurobiologically plausible way. Therefore, I designed the encoder such that it adheres to properties found in cortical spiking networks, i.e., it encodes information in spikes with neurobiologically plausible spike rates (Markram et al., 2015) which is injected in a distributed fashion that resembles spike patterns generated by a Poisson process (Shadlen & Newsome, 1998). I investigate the effect of different encoding schemes on processing and discuss different proposals for the neural code in chapter 3. Based on these results, I use a scheme with a temporal code where information is encoded solely through the precise timing of the input spikes. This spike-based encoding corresponds to the way cortical neurons communicate and delivers good performance on fundamental tasks such as input separability and memory retrieval (see chapter 3). Furthermore, the encoding scheme does not use a spatial code where different input stimuli are projected to different neurons. Also, in contrast to a rate-based code, the firing rates in the temporal code were kept constant. This encoding represents a minimalist version of a temporal code that does not rely on additional assumptions. Similarly, I use a featureless encoding where the input does not contain information beyond word identity. In other words, the word encodings have no internal structure, such as syntactic or semantic features, and this serves as an illustration of what is possible without specific assumptions about feature structure.

2.1.4 Readout

To characterize the output of a processing task, I use the readout component. It reads the network state and uses it to determine a solution to a given task. The readout here is a calibrated measurement device that serves as a stand-in for a down-stream neurobiological system, with adaptive characteristics that in reality would readout the activity of the NBL model network. It identifies the aspects of neural activity that are relevant to solve a given task and extracts the corresponding information via neuronal activity. This allows for testing whether the relevant information to solve a task is available in the network or not. If the NBL model as a whole can *not* solve a task, the relevant information is not provided by the neural network. Conversely, if it can, then the relevant information is available in the state of the network and could therefore also be accessed by downstream networks.

The readout is simple and limited to linear operations on the momentary state of the network. Hence, any processing step that requires memory or non-linear operations on the input is not performed by the readout. Therefore, if the NBL model can solve a task requiring processing memory or non-linear integration, the relevant processing is performed by the neural network. Since the readout is used as a measurement device and the readout weights do not affect processing in the network itself, several readouts can be attached in parallel without interfering with the network, or each other, each reading out different types of information in different tasks. Thus, the information in a single network can be used to solve several tasks in parallel and in this sense the NBL model is accomplishing parallel processing related to various computational needs. I use this feature, for example, in chapter 6, where I use parallel readouts to retrieve different aspects of each sentence through a set of readouts operating on the same network state. This includes word identities of different sentence constituents and the event semantics of the sentence.

Readout from neuronal state, not spikes

To determine an output, the readout is measuring the neuronal states, i.e., the momentary membrane potential u and adaptive current w of each neuron (see Figure 2.2). This is somewhat unusual in computational neuroscience where a readout often uses a temporal filter applied to the spikes generated by the neurons. This is often done since biological neurons communicate through spikes. Therefore, such is the reasoning, the information contained in the neuronal state is not necessarily accessible to downstream networks. Instead, I apply the readout directly on the neuronal state for two reasons: First, the information extracted with the readout can indeed be made available to a downstream readout network. This can be done for a network and any connectivity structure by copying the neurons and all of their incoming spikes to the readout network as illustrated in Figure 2.3. There we can see that, instead of the neuronal spikes generated by the neurons targeted by the readout (blue and orange), the spikes arriving at the targeted neurons are transferred to the readout network. This reproduces the state of the neurons that the NBL readout has access to in a hypothetical readout network. Thus, the information that is available to the readout (e.g., membrane state) is also accessible by downstream networks even though neurons only communicate through spikes.

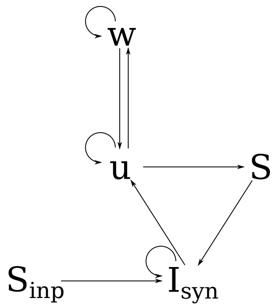


Figure 2.2.: Interaction of different neuronal and network state variables in the NBL model. The input spikes S_{inp} generate synaptic currents in the receiving neurons I_{syn} that change the membrane potential u. The membrane potential of each neuron interacts with the adaptive current w of the neuron. When reaching the firing threshold, the membrane potential elicits spikes S which, in turn, generate synaptic currents in the post-synaptic neurons I_{syn}. These, in turn, change the membrane potentials of the post-synaptic neurons and so on. Each of the state variables u, w and I_{syn} influences their own future state, indicated by the looped arrows.

The second reason why I read out from the neuronal state relates to the question of how a readout based on spikes is implemented. Spikes are short-lived and are modeled as events without temporal extension. Thus, the integration of several spikes requires some convolution kernel, a temporal filter that extends the effect of a single spiking event over time. This generates a time-continuous signal that can then be used by a readout. However, what properties should this temporal filter have? If, for example, it would have a long time constant, it could retain information longer than the network itself. The ideal temporal filter is one that does not omit or add information to what is currently available in the network. The best filter that satisfies these conditions is a neuron itself (van den Broek et al., 2018). A neuron receives spikes and represents them in its neuronal state as a time-continuous signal, i.e., it has the properties of a temporal filter. Directly reading out the neuronal states obviates the need for a spike filter with arbitrary time constant. In addition, there is also no need to introduce extra readout parameters that may require testing and tuning. Therefore, neuronal states are read out directly from the NBL model without loss of generality.

Biological plausibility of the readout

One of the limitations of learning in biological networks is that information is only available locally, either within the neuron itself or through communication with its neighbors. The estimation algorithm to determine the readout in the NBL model,

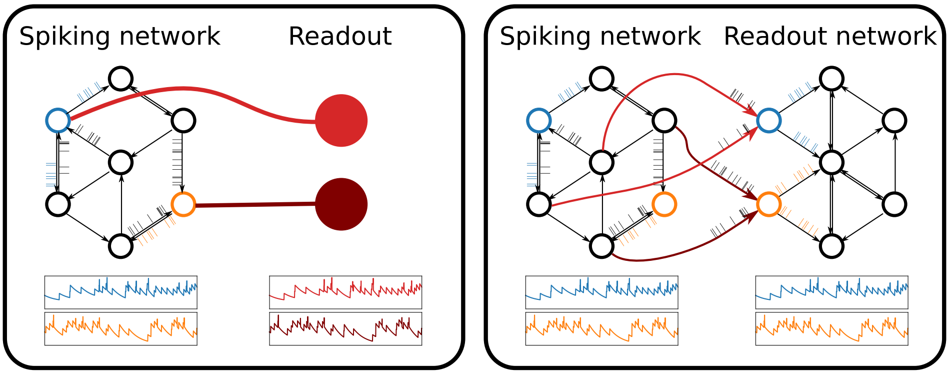


Figure 2.3.: Illustration how the information available to the readout can be transmitted to a readout network. The left panel shows how the readout is accessing the membrane potentials of two neurons in the NBL model. The right panel shows how incoming spikes to the blue/orange neurons are transferred to a downstream readout network such that the membrane activity of both neurons is recreated in this readout network. Hence, all information that the readout in the NBL model has access to is also available to a downstream network.

linear regression, uses non-local information. It has access to the state of all neurons at once and hence to the global network state. However, after training, the readout only uses linear operations to extract information from the network, i.e., information from different neurons is only scaled and added. This can plausibly be done by downstream readout networks as well. Furthermore, parallel readouts are also implemented in neurobiology with multiple downstream networks connecting to different subsets of neurons in an upstream network. Each downstream network can thus extract relevant information for its needs without interfering with other readouts.

2.1.5 Scientific embedding

Memory, Unification, Control

Conceptually, I work within the framework of the Memory, Unification & Control model (Hagoort, 2005) and the NBL model aims to implement aspects of unification. The encoder provides information that would be stored in long-term memory, the mental lexicon. This information is retrieved and delivered as spike patterns to the unification network. To base the model on as few assumptions as possible, these spike patterns do not contain information about lexical features of the words presented, including syntax and semantics. Instead, the retrieved word representations only encode word identity. The network in the NBL model is stimulated by the input which it then processes in the context of previous input. If the readout can identify sentence-level properties, e.g., semantic relations, based on the network state, this shows that the network can integrate information provided by the encoder to generate larger conceptual structures, i.e., it performs unification. If the readout is successful this implies that the result of unification can be delivered to downstream networks. These could further process the information to perform more complex cognitive tasks involving, for example, the larger linguistic context, or reasoning based on the sentence-level interpretation.

Reservoir computing

The design of the NBL model is similar to reservoir computing models with spiking neurons. Reservoir computing (Lukoševičius & Jaeger, 2009) describes a class of neural architectures that consist of a randomly and statically connected network (the reservoir), an input, and a trained linear output. There are several different network types that fall within the class of reservoir computing. These include the liquid state machine (Maass et al., 2002) and echo state networks (Jaeger, 2001) that differ from each other in terms of the neuron model they used. In echo state networks, the aim was to build a system that reproduces the behavior of an irregular (chaotic) oscillator. To do this, the network was trained to predict the state n+1 of a chaotic system when given state n. Then, the input and output nodes were connected such that the network became a closed system, using non-spiking computational units.

The NBL model is more similar to a liquid state machine that uses spiking neurons and has the purpose to perform computations on an input presented to the network. However, the NBL model relies on more complex two-dimensional, adaptive neurons which results in more complex network dynamics. Research into reservoir computing as a machine learning tool illustrates the computational potential of the NBL model (Lukoševičius et al., 2012). However, in the current work, I have a different explanatory goal: I am using a readout on the states of a spiking network to investigate neuronal computation and the functional role of different parameters and neurobiological features, rather than as a machine learning tool.

2.2 General Methods

2.2.1 Neural network

Neuron model

In my network simulations I used the *adaptive exponential neuron model* (AdEx; Brette & Gerstner, 2005) which is based on neurophysiological experiments to mimic the individual neuron response to input current stimulation. The AdEx is a two-dimensional neuron, where one dynamical variable models the membrane potential u while the other, w, is a spike-triggered current. This current is used to model adaptive processes in the neuron, for example spike rate adaptation. The behavior near the spike threshold is governed by an exponential term which fits biological neurons better than models with a linear behavior near the threshold (Gerstner et al., 2014).

In my implementation, the dynamics of the neuron model is defined by the following two equations:

$$C_m \frac{d}{dt} u(t) = -g_L(u(t) - E_L) + g_L \Delta \exp\left(\frac{u(t) - V_t}{\Delta t}\right) - w(t) + I(t)$$

$$\tau_w \frac{d}{dt} w(t) = a(u(t) - E_L) - w(t)$$

Both dynamical variables have their own decay time constants. For the membrane potential u this is $\tau_u = \frac{C_m}{g_L}$, with C_m being the neuron capacitance and g_L the leak conductance. For the adaptive current w this time constant is τ_w. Other parameters are the reversal potential E_L, the spike threshold V_t and a which determines the interaction between u and w. Δ governs the influence of the exponential term, see Table 2.1 for parameter values.

When u surpasses the peak potential V_p, a spike is triggered. V_p has to be significantly higher than V_t, however, the exact value is not critical. Crossing V_p indicates

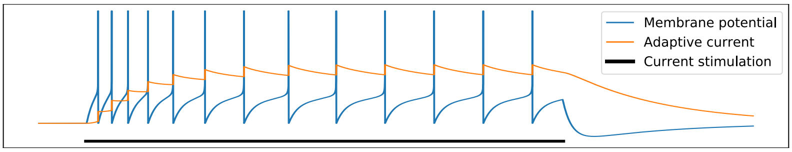


Figure 2.4.: Temporal evolution of dynamical variables in the AdEx neuron. Illustration how the membrane potential u and the adaptive current w change due to a constant step current as input. The black line indicates when the input current is active. Due to the adaptive current, the spike rate decreases over time despite the constant stimulation.

Name	**Value**	**Description**
C_m	$0.45\,\mathrm{nF}$	membrane capacitance
g_L	$25.0\,\mathrm{n}\Omega^{-1}$	leak conductance
Δ	$2.0\,\mathrm{mV}$	rise slope factor, influence of exponential term
E	$-70.6\,\mathrm{mV}$	leak reversal
V_t	$-50.4\,\mathrm{mV}$	threshold potential
V_p	$20.0\,\mathrm{mV}$	peak potential of exponential regime
V_r	$-70.6\,\mathrm{mV}$	reset voltage
a	$4.0\,\mathrm{n}\Omega^{-1}$	sub-threshold adaptation conductance
b	$80.5\,\mathrm{pA}$	increment of w on spike emission
τ_w	chapter dependent	adaptation time constant
τ_m	$18.0\,\mathrm{ms}$	membrane time constant

Table 2.1.: Parameters used for AdEx neuron model.

that the membrane potential has entered the exponential regime of the dynamical system described by the model equations. So, V_p does not have a neurophysiological interpretation and changing it only shifts spike positions by fractions of a millisecond (Clopath et al., 2007). At the firing time t_f, the neuron parameters are updated to model the spiking process:

$$\begin{aligned} u(t_f) &\leftarrow V_r, \\ w(t_f) &\leftarrow w(t_f) + b. \end{aligned}$$

V_r is the reset potential and b determines how strong the adaptive current w is increased by a spike. The main parameters that determine the firing characteristics of the neuron are a, b, V_r and τ_w (Naud et al., 2008). By manipulating them, the AdEx model can mimic different electrophysiological classes of neurons. I used parameters similar to the original publication (Brette & Gerstner, 2005) that showed spike rate adaptation, as do the majority of excitatory cortical neurons (Markram et al., 2015). This process is depicted in Figure 2.4.

Synapse model

To model a synapse, I simulated an exponentially decaying current which was increased instantaneously when the pre-synaptic neuron spiked, for both excitation and inhibition of the post-synaptic neuron. The dynamical variable $I_i(t)$ represents the momentary sum of all synaptic currents to neuron i. $I_i(t)$ was convolved with a Dirac delta function (see glossary, section 2.4) corresponding to each spike time. In other words, the current $I_i(t)$ was increased by a fixed amount for every pre-synaptic spike onto neuron i. The amount was determined by the synaptic weight w_{ij} from the pre-synaptic neuron j to the post-synaptic neuron i. Formally, it was defined by the differential equation:

$$\tau_{syn} \frac{d}{dt} I_i(t) = \left(-I_i(t) + \sum_j w_{ij} S_j(t) \right) \cdot I_{\text{app}}$$

where τ_{syn} was the synaptic time constant and $S_j(t)$ a series of delta functions representing the spike train generated by neuron j. I_{app} was a global scaling parameter to tune the network activity (see next section). In chapter 4, I investigate how different values for τ_{syn} influenced the memory characteristics of the network. There was no synaptic plasticity in the NBL model.

Network connectivity

The network consisted of 1000 neurons, 200 of which were inhibitory and 800 excitatory. A synapse between two neurons was realized with a probability of ρ_c. I investigate the effect of different values for ρ_c in chapter 4. The strength of the synaptic connections that were realized was defined by a unit-less scaling weight. The weights of the excitatory synapses were independently drawn from a uniform distribution between 0 and 0.2 while inhibitory synapses were independently drawn from a uniform distribution between 0 and 1. The ratio of inhibition and excitation reflected the distribution in cortex and approximates the ratio in a mammalian layer 2/3 network (Markram et al., 2015). The relative strength of inhibitory and excitatory weights provided an approximate balance between excitation and inhibition (Tsodyks & Sejnowski, 1995).

Numerical simulation

I used the forward Euler scheme (see glossary, section 2.4) with a time step of 1 ms to integrate the differential equations defining the neuron model and simulate the network. Generally, simulations were repeated ten times with different random seeds which determined the network connectivity, input spike patterns, and the training and test sets. I refer to simulations with different seeds as network subjects.

Before the start of each simulation, a scaling parameter I_{app} was tuned iteratively until the network displayed a mean spike rate of f_{tune_n} across neurons on the first 500 stimuli of the input set within a 10% margin of the tuning rate f_{tune_n}. I test how different values of f_{tune_n} change the network properties in chapter 4. There was no background activity stimulating the network. Figure 2.5 shows typical evoked activity during input presentation. Except for chapter 3, the network state was reset before each input sentence since the focus of this work was on single-sentence processing. The state to which the network reset was determined at the end of the rate tuning procedure described above. The reset state included the dynamic variables u, w and I for each neuron and the network was restored to the saved values between sentences. To investigate discourse processing, one would disable reset to allow information flow across sentence boundaries.

2.2.2 Input encoding

The input encoder assigned a fixed, randomly generated spike pattern for each word to stimulate the network. The patterns consisted of different instantiations of 20 Hz Poisson processes on 100 input channels. Each of the channels was connected to a fixed set of 5% randomly chosen excitatory neurons in the network and was active for every word. Thus, word information was exclusively encoded in the timing of spikes on the input channels. When words were repeated they were represented by the same temporal spike pattern (frozen noise). The input weights were excitatory and chosen from a uniform distribution between 0 and 0.2. There was also a global input weight scaling parameter I_{inp} which was tuned such that the input drove a spike rate of 2.5 Hz, averaged over all neurons when all network-internal connections were set to zero. This ensured that the input stimulation had a consistent strength across simulations, and was independent of the strength of the internal, recurrent connections.

The duration of each stimulus was determined by multiplying the orthographic length of the word with 50 ms. For example, the longest word was 9 letters long, represented by a spike pattern of 450 ms in duration. The shortest word was represented by a 50 ms spike pattern. The mean duration over all 212 different words used in the language was 250 ms with a standard deviation of 70 ms. The words were delivered sequentially without pause (Figure 2.6). The randomly chosen spike patterns, the constant spike rate per input channel and the non-localist projection to the network in combination have the effect that there were no clear word boundaries in the input patterns. The input was indistinguishable from continuously generated Poisson noise with a rate of 20 Hz and information was provided solely through the unique spike pattern that characterized each word in the language.

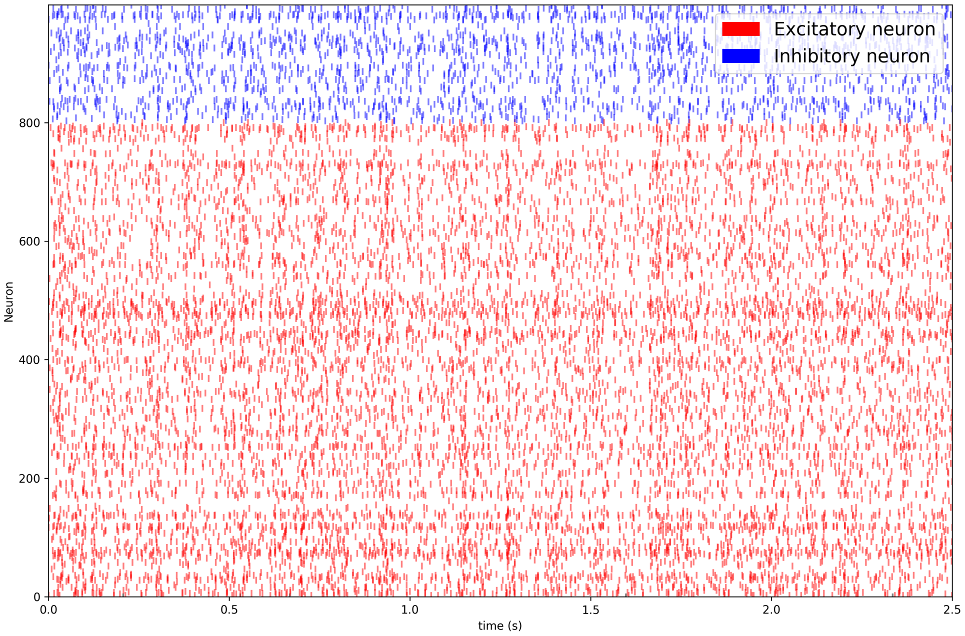


Figure 2.5.: Spike raster plot of the network. Activity in the model while receiving input with the network parameters from section 4.3.4. The input patterns that were used are shown in Figure 2.6.

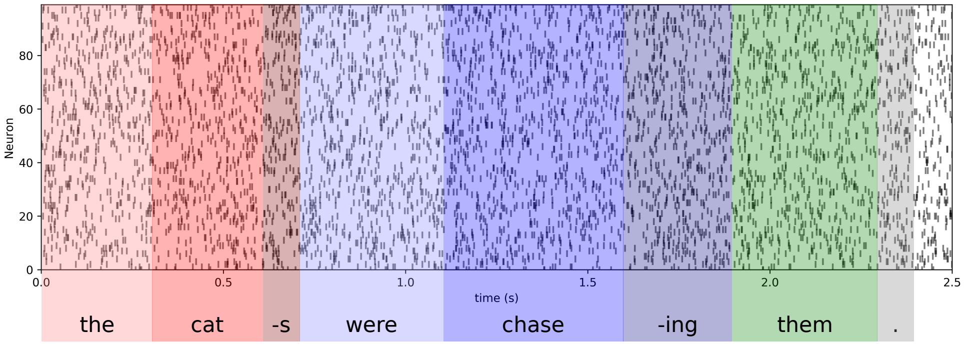


Figure 2.6.: Spike patterns for encoding of a sentence. Example of how a sentence is encoded as a spike pattern on 100 input channels. The colored patches indicate word boundaries in the spike pattern. There was no change in spike frequency or the number of activated neurons at word boundaries. The duration of each word pattern corresponded to the orthographic length of the word with 50 ms per letter.

2.2.3 Readout

The computational tasks and target outputs varied across chapters. There is (delayed) stimulus identification in chapter 3, semantic role assignment in chapters 4 and 5 and query answering. In chapter 6, all of these tasks are performed in parallel using different readouts. To produce outputs, I trained linear readouts on the network activity. I used static networks without synaptic plasticity and only the connections of the network to the readout were task specific. The readout weights were estimated using ordinary least squares regression (glossary, section 2.4) on the neuronal state variables u and w. Both variables were sampled every 5 ms, standardized per neuron to a mean of 0 and standard deviation of 1, and then averaged by stimulus. That is, there was one data point per neuron and state variable for each stimulus.

Each task had several possible readout values (e.g., different semantic roles) and I used a winner-takes-all approach. Thus, the readout value with the highest activation given the stimulus-averaged network state was selected. This determined the output assigned on the current input, e.g., the semantic role that was assigned to a word. The tasks varied in terms of difficulty. Because of that, the size of the training set n_{train}, i.e., the number of words used to estimate the readout weights, and the size of the test set n_{test}, i.e., the number of words used to determine the performance of the readout, were chosen differently in each chapter and are reported in the methods.

Random classifier

As a baseline to compare the performance of the NBL model against, I used a random classifier in chapters 4, 5 and 6. I took the target output for the test set, randomly permuted the order to generate a random assignment, and then computed its performance. In this way, the distribution of the different output values over the test set was maintained. Thus, if some target outputs were more frequent than others, this was reflected in higher performance of the random classifier compared to when all targets were equally likely. I used this random baseline to compute a conservative kappa score for multinomial classification to assess the different models (glossary, section 2.4).

Back-off-N-gram model

In chapter 4, I compared the NBL model to a back-off N-gram model as a benchmark. It first collects all possible N-grams from the training set, i.e., all chunks of N contiguous words, for all N up to sentence length. For example, the sentence “cats chase dogs" would be chunked into the set {"cats", "chase", "dogs", "cats chase", "chase dogs", "cats chase dogs"} of N-grams. Then, the frequency of each semantic role for the last word in each N-gram was calculated and stored. For instance, the trigram "the big cat" might occur 10 times in the training set and in 8 of these cases *cat* was the AGENT while in the other 2 cases it was a PATIENT. This information was recorded in memory for all chunks in the training set.

In testing, the back-off N-gram model processed sentences one word at a time. At each word position it would try to find the N-gram of the whole sentence up to and including this word in its memory. If the N-gram did not exist, it would delete the initial word in the N-gram and try again. For example, when processing "the big cat chased dogs" sequentially, at "dogs" it would first look for "the big cat chased dogs" in memory, then "big cat chased dogs", then "cat chased dogs", and so on. Once, the largest piece of context was found that existed in memory, it assigned the most frequent role to the final word (see above).

This model has perfect memory of its input history and access to all N-gram information in the training set. It is interesting for comparison with the NBL model-because it uses the largest contextual chunk that is compatible with experience and selects the most probable role based on frequency. However, it cannot generalize beyond lexical sequences.

2.3 Input language and semantic roles

As model input, I used an English-like artificial language. The basic version of this language is described here, some adaptations that were made are described in the respective chapter methods.

Input sentences were generated using a construction grammar (Goldberg, 2006). The language had six basic constructions and six syntactic alternations. Each of the construction templates could generate a number of different sentences depending on the phrase semantics and the possible lexical items that filled the template slots. The lexicon size was 212 words and, in total, the grammar could generate around four hundred billion distinct utterances. Constructions in the language had one verb and up to three noun phrases plus function words. The semantic roles of noun phrases were determined by the construction type. Possible semantic roles in the language were AGENT, PATIENT, THEME, EXPERIENCER, RECIPIENT, GOAL and INSTRUMENT. The semantic role of auxiliaries, verbs and verb inflections was set to ACTION. Table 2.2 shows all construction templates that were used.

Construction (frequency)	**Word categories and examples**
Inanimate Intransitive	Action = ERGATIVE; Patient = OBJECT
Main form (9,5%)	*A cup was break-ing.*
Animate Intransitive	Action = UNERGATIVE; Agent = LIVING
Main form (9,5%)	*Old man-s jump-ed.*
Transitive (Agent-Patient)	Action = TRANSITIVE (AGENT-PATIENT); Agent = LIVING; Patient = OBJECT; (Instrument = INSTRUMENT)
Main form (19%)	*The man/dog catch-s^{3rd} the small apple.*
Syntactic alternation (9,5%)	*The small apple/dog is catch-ed^{par} by her.*
Instrumental preposition (5%)	*The man catch-s^{3rd} the small apple with a net.*
Alt. instrumental prep. (2,5%)	*The apple is catch-ed^{par} by the man with a net.*
Transitive (Theme-Experiencer)	Action = TRANSITIVE (THEME-EXPERIENCER); Theme = OBJECT; Experiencer = LIVING
Main form (9%)	*The stick/cat scare-ed the man.*
Syntactic alternation (5%)	*The man/cat was scare-ed^{par} by a stick.*
Transfer Dative	Action = DATIVE; Agent = LIVING; Theme = OBJECT; Recipient = LIVING
Prepositional object (9,5%)	*The man give-s^{3rd} the apple/mouse to a woman/cat.*
Double object (9,5%)	*The man give-s^{3rd} the woman/cat the apple/mouse.*
Locative	Action = LOCATIVE; Agent = LIVING; Goal = LOCATION; (Instrument = INSTRUMENT)
Main form (9,5%)	*A woman was go-ing to the table.*
Instrumental preposition (2,5%)	*A woman was go-ing to the lake with a bike.*
Roles:	AGENT PATIENT THEME EXPERIENCER RECIPIENT GOAL INSTRUMENT ACTION

Table 2.2.: Constructions in the input language. Each section of the table describes one construction and its syntactic alternations (if any). The percentages show how often each form occurred in the language. The right column shows which word categories from table 2.3 were used and an example to each of the syntactic forms. The colors in the example sentences refer to the semantic roles of words, indicated on the bottom of the table. The alternative nouns in the examples (marked with a slash) indicate the positions where semantically unconstraining nouns can replace the semantically constraining nouns. Unconstraining nouns can assume an active and a passive role in a sentence. Thus, they introduce local ambiguity at the sentence beginning which does not exist for constraining nouns.

The frequency of the different constructions was chosen to broadly reflect distributional properties of English. Generally, active forms of a construction were generated twice as often as the passive forms. Other alternations were generated less frequently. I assumed the transitive construction to be more common than the others and hence gave it higher probability, see Table 2.2 for the actual numbers.

Verbs were construction-specific, i.e., verbs from each category could only occur in one construction and its syntactic alternations. There were 16 different items in each verb category, see Table 2.3. The nouns were not construction-specific but for each noun phrase slot in a construction, only certain nouns were allowed depending on their semantics. LOCATION and INSTRUMENT nouns could only be in noun

Word category	**Examples**
Determiner	the, a
Pronoun	he, she, it, him, her, they, them
Noun LIVING (constraining)	man, woman, father, mother
Noun OBJECT (constraining)	ball, cake, orange, banana
Noun LIVING/OBJECT (unconstraining)	cat, dog, rat, mouse
Noun LOCATION	lake, beach, tree, roof
Noun INSTRUMENT	net, knife, bike, stick
Adjective	young, old, big, little
Verb ERGATIVE	sleep, dance, jump, sit
Verb UNERGATIVE	open, close, break, smash
Verb TRANSITIVE (AGENT-PATIENT)	kick, chase, lift, follow
Verb TRANSITIVE (THEME-EXPERIENCER)	scare, surprise, hurt, bother
Verb LOCATIVE	go, walk, drive, run
Verb DATIVE	give, throw, show, present
Auxiliary Verbs	is, are, was, were, being
Prepositions	to, by, with

Table 2.3.: Word categories. Each noun, verb and adjective category had 16 lexical items. These word categories were used to instantiate the different constructions in Table 2.2. Auxiliary verbs were used to express tense and aspect. The preposition *to* was used in the locative and transfer dative constructions, *by* occurred in passive voice and *with* was used for instrumental alternations.

phrases with the semantic roles GOAL and INSTRUMENT, respectively. Nouns from the category LIVING could only occur in noun phrases with the roles AGENT, RECIPIENT or EXPERIENCER. Nouns from the OBJECT category could only be used in noun phrases with the roles PATIENT or THEME. The LIVING/OBJECT noun category was special because these nouns could occur in noun phrases that instantiated both the LIVING or the OBJECT category. This creates semantically unconstraining nouns that, for example, could either be AGENT or PATIENT of an action. Examples of this noun category are animals which can both be the agent, as in *The dog chases the ball.*, or the patient, as in *The cat is caught by the man.* These nouns are interesting to investigate with respect to the integration of semantic cues (see chapter 5). I only used this category in selected constructions and positions. In Table 2.2 these positions are marked by alternative nouns in the examples (such as *man/dog*). In these slots, constraining and unconstraining nouns were equally likely.

Noun phrases consisted of a determiner, an adjective, and a noun. The determiner was definite (*the*) with 60% probability and indefinite (*a*) with 30% probability. In 10% of all cases, the noun phrase was replaced by a pronoun with the corresponding gender and number. With 75% probability, the noun phrase was singular and otherwise plural. In accordance with English grammar, an indefinite determiner was

Grammatic context ***example***	**Event semantics**	**Example verb phrase**
Active sentence, singular subject	Simple present	*chase -s*^{3rd}
The cat chase -s^{3rd} *the ball.*	Simple past	*chase -ed*
	Progressive present	*is chase -ing*
	Progressive past	*was chase -ing*
Active sentence, plural subject	Simple present	*chase*
The cat -s chase the ball.	Simple past	*chase -ed*
	Progressive present	*are chase -ing*
	Progressive past	*were chase -ing*
Passive sentence, singular subject	Simple present	*is chase -ed*^{par}
The ball is chase -ed^{par} *by the cat.*	Simple past	*was chase -ed*^{par}
	Progressive present	*is being chase -ed*^{par}
	Progressive past	*was being chase -ed*^{par}
Passive sentence, plural subject	Simple present	*are chase -ed*^{par}
The ball -s are chase -ed^{par} *by the cat.*	Simple past	*were chase -ed*^{par}
	Progressive present	*are being chase -ed*^{par}
	Progressive past	*were being chase -ed*^{par}

Table 2.4.: Verb constituents were adapted by subject number and event semantics. List of all possible variations of the verb constituents for the example *chase*. Auxiliaries and verb suffixes depended on subject number, aspect, tense and voice.

removed if the noun phrase was plural. In each noun category, there were 16 different lexical items. The adjective category also included 16 items and was not specific to different noun categories, i.e., no information about a noun could be inferred from the adjective. Each adjective could occur in phrases with any semantic role. In 75% of all noun phrases, the adjective was omitted.

Each sentence had a randomly chosen tense (present or past) and aspect (simple or progressive) which amounted to 4 different combinations, all of which were equally likely (see Table 2.4 for an example). Inflected words such as verbs in passive, past or progressive forms or plural nouns were marked with morphemes "-ed", "-ed^{par}", "-ing", "-s^{3rd}" or "-s". These were treated as a separate words by the input encoder. Here, "-ed^{par}" stands for the passive marker to differentiate it from the past tense marker and "-s^{3rd}" stands for the third-person-s to differentiate it from the plural-s. The construction grammar generated sentences with different word lengths and durations; from two-word sentences such as *They sleep.* with a length of 450 ms, up to eighteen-word sentences such as *The young women-s were being chase-ed*^{par} *by the angry father-s with the little bike-s.* with a length of almost 4s. Each of the sentences in the training and test sets was unique. Repeating content words were excluded, i.e., sentences like *The black cat is chase-ing a red cat.* were discarded.

2.4 Mathematical glossary

Throughout this work, I rely on some mathematical terms to describe the NBL model. These terms are standard in computational neuroscience and I introduce them in alphabetical order.

Dirac delta pulse

A delta pulse $\delta(x)$ is a mathematical concept that is used to describe idealized point masses, point charges, or in this case, spike events which are "points" in time. It is zero everywhere except at point x and its integral is defined as $\int_{-\infty}^{\infty} \delta(x)dx = 1$. Since my simulations used discrete time, delta pulse events were implemented as acting for exactly one time step, independent of its length.

Forward Euler method

The Forward Euler method is a numerical procedure for solving an ordinary differential equation $\frac{dx}{dt} = f(x)$ with initial value $x(0) = x_0$. It is a first order method that approximates a curve by a series of tangential lines whose slopes are determined by $f(x_i)$ at the integration points x_i. The quality of the approximation depends on the step size $h = x_{i+1} - x_i$. In all simulations I set $h = 1ms$. I used this method to approximate the evolution of the dynamical equations describing neurons and synapses and their interactions in the network.

Gaussian white noise

Gaussian white noise ξ is a stochastic process characterized by its expectation value $\text{mean}(\xi(t)) = 0$ and standard deviation, with realizations that have no temporal autocorrelation: knowledge of the value of ξ at time t does not allow prediction at any other time t'$\neq$t. I used it as discrete-time noise process that is generated by a random draw from a Gaussian distribution with mean 0 and standard deviation σ.

Kappa correction

Readout performance was kappa corrected to factor out what a random classifier could achieve. This was done by projecting the raw performance P_{raw} of a readout from the interval between 0% and 100% to the interval between the score of a random classifier P_{rand} and the maximally possible score P_{MAX}. This yields the kappa score P_κ.

$$P_\kappa = \frac{P_{raw} - P_{rand}}{P_{MAX} - P_{rand}}$$

The score P_{MAX} was task-dependent and could be 100% or less. It is indicated each time kappa correction was used.

Linear regression using ordinary least squares

Linear regression is a method to find a solution for the equation $\mathbf{y} = \mathbf{X}\,\mathbf{b} + \mathbf{r}$ where $\mathbf{y}$ and $\mathbf{X}$ are given and $\mathbf{r}$, the error term, should be as small as possible, by some measure. This means the equation needs to be solved for $\mathbf{b}$ in a way that fulfills the minimal error condition. The ordinary least squares method provides a solution that minimizes the square of the error $\mathbf{r}$ by applying the formula $\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1} \cdot \mathbf{X}^T\mathbf{y}$. In the NBL model, the matrix $\mathbf{X}$ corresponds to the aggregated network states (u and w for each neuron) over time and the vector $\mathbf{y}$ corresponds to the target output at each point in time. The estimated values for $\mathbf{b}$ were then used as weights for the connections from the network to the readout.

Poisson process

A Poisson process is a stochastic process that generates random occurrences of an event that happen within a given time window. The probability P of an event being realized n times in a given time t is $P = \frac{(\lambda t)^n}{n!} e^{-\lambda t}$ where the rate λ is the sole parameter defining the statistical process. Poisson processes have been found to approximately describe the distribution of cortical spikes (Softky & Koch, 1993), so I used them to generate input spike trains from sources external to the network. In simulations, I approximated the Poisson processes by allowing only one event per time step, that is, I only considered $n = 1$ for a time step of $dt = 1ms$ and omitted all additional spikes. The probability of $n \geq 2$ events happening with a spike rate of $\lambda = 20\,\text{Hz}$, which is the rate used for most of the input stimuli, was smaller than 0.02%.

3

Effects of input encoding on processing and memory

3.1 Introduction

In order to generate sentence-level interpretations in a context-sensitive manner, the human brain has to implement structured sequence processing capabilities in neurobiological infrastructure (Petersson et al., 2012). One way to identify possible mechanisms for this is to build neurobiological models and investigate the sequence processing capacities in these models. Simulating neural populations of various sizes and degrees of realism is an active field of research. Particular focus has been on studying the different dynamic regimes of network activity (Ostojic, 2014; Wang, 2002; Rabinovich et al., 2001) and the influence of different model features, including connectivity (Wallace et al., 2013), neuron (Izhikevich, 2003; Brette & Gerstner, 2005) and synapse model (Duarte & Morrison, 2014; Zenke et al., 2015), on the network dynamics. However, it is less well-researched how the computational properties of networks are influenced by the manner in which information is encoded into these networks.

The spiking activity of networks depends on how information is delivered as input (Duarte & Morrison, 2014; Duarte et al., 2018). For example, if only one neuron is stimulated externally, it will be activated and activate the neurons it is directly connected with. But other neurons in the network will be less active. If, in contrast, all neurons are stimulated equally, the network will show more homogeneous activity and all neurons fire with similar spike rates. Even though the overall spike rate might be the same in the two situations, the nature of the network activity is different. This is illustrated in Figure 3.1. Similarly, the way the neurons are stimulated changes

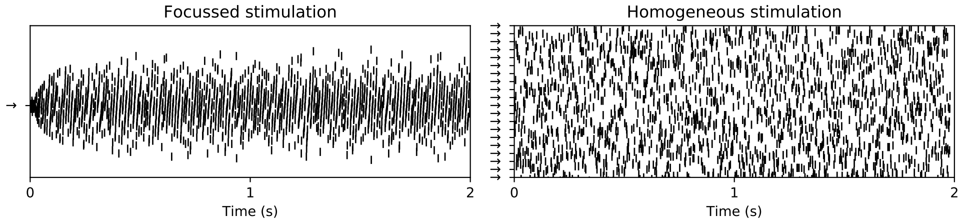


Figure 3.1.: Illustration how stimulating a single neuron (left, arrow) generates a different type of activity compared to a network where all neurons are receiving an equal stimulation (right, arrows) while the total spike-rate is the same in both cases. For illustrative purposes, the network connectivity is arranged in a line such that neurons are only connected to neurons that are directly above or below them in the spike raster plots.

the spiking behavior. If a single neuron is stimulated with a constant current, it will typically spike regularly or with adapting inter-spike-intervals. But if it is stimulated by a spike train generated by a Poisson process, it will show more variable behavior with interleaved short and long inter-spike-intervals depending on the spike times in the input. Thus, it is conceivable that the information processing taking place in a network depends on both how the network is stimulated, the encoding scheme, and on which neurons are stimulated, the input projection. For example, different stimuli may be easier or harder to distinguish, depending on these factors. If in addition, the neurons are equipped with dynamic memory variables, the network might retain information for a longer period with one input regime compared to another (Petersson, 2005).

To test the influence of input encoding on processing, I compared versions of the NBL model with three different encoding schemes, using two different input projections, on three benchmark tasks.

3.1.1 Encoding schemes

The first encoding scheme stimulates neurons with *direct currents* which is inspired by patch-clamp experiments in neurophysiology (Rauch et al., 2003). In these simulations, a constant input current generates spiking in a subset of neurons that then drive population activity (e.g., Boerlin et al., 2013; Ostojic, 2014). Since biological neurons largely communicate through spikes, direct current stimulation does not represent a natural mode of signalling but implements a reference model that allows noise-free information transfer to neurons.

The other encodings reflect two possibilities of how cortical neurons communicate. In the debate on how information is coded in spike trains – the question of what the neural code is – there are typically two positions. One side argues that the timing of spike events carries information (temporal code perspective), while the other side views the firing rate of neurons as the carrier of information (rate code perspective). One of the earliest experiments investigating the neural code found that stronger tension of a muscle leads to higher firing rates of the sensory neurons in the muscle (Adrian, 1926). Later, Rieke and Warland (1999) argued for a temporal code based on the fact that it supports higher coding efficiency compared to a rate code (see also Thorpe et al., 2001) and the finding that response times to a stimulus can be of the same order of magnitude as the inter-spike intervals of the neurons processing the stimulus. This suggests that a single, or a few spikes, can encode response relevant information and supports the temporal code view. These results are consistent with experimental findings on sound localization in the barn owl that found that a temporal code allows the animal to perceive time differences of less than 5 ms (Gerstner et al., 1997).

However, others have questioned the interpretation of these findings. Borst and Theunissen (1999) argue that the fact that single spikes matter is not evidence against the rate code since the time windows over which a rate is computed can be arbitrarily small. Hence, sensitivity to single spikes can support both, rate and temporal code models. London et al. (2010) find that single spike suppression in the rat barrel cortex can have large effects on population activity. In combination with the observed variability of spike timings in neurons, they argue that this supports a neural code that is robust to perturbations, suggesting a rate code over populations of neurons. In addition, Ahissar et al. (2000) found that a temporally coded signal from rat whisking is translated into a rate code in the rat's cortex which suggests the coexistence of both neural codes. According to them, either coding scheme can be used by different systems depending on the application.

More recently, Brette (2015a) has taken a different approach in this discussion. According to Brette, the question “Does the brain use rate or temporal coding?” is ill-posed. A better question would be “Is it possible to reduce the spiking interactions of neurons to the interaction of rates?”. He argues that rate coding is a summary statistic derived from measurement methodologies while spikes are the biological basis of neuronal communication and live in the time domain, i.e., they are causal. He rejects the idea that both coding schemes are extremes of a continuum, or that brains use a mix of the two. He argues that correlations between spike rates and information processing (like Adrian, 1926) is not evidence against the temporal code, since also in a temporal code spike rates can fluctuate and spike timing can carry the same information.

In this study, I use one version of a temporal neural code to deliver information to the neural network, which I refer to as *temporal encoding scheme*. I stimulate the network with spike trains generated by a Poisson process with a fixed constant rate. Each stimulus is encoded through different Poisson processes with the same rate such that only the instantiation of the Poisson process, the precise spike times, encode information. These spike trains are delivered over several input channels that each have different connections to the network. Following the analysis of Thorpe et al. (2001), this encodes information using both latency (spike timing) and rank (spike order) information.

I also use a *rate encoding scheme* and compare it with the temporal code and direct current injection. I stimulate networks with spike trains where the precise spike position varies between presentations of the same stimulus. However, the spike rate, a parameter of the Poisson processes generating the spike trains, can be used to identify the different stimuli. Again, I use several independent input channels that each have a different but constant rate per stimulus. Therefore, different stimuli can be identified by the combination of spike rates on the individual input channels. The three encodings are shown in Figure 3.2.

It is worth noting that network models that do not use spiking neurons implicitly assume a form of rate-based coding. In computational neuroscience, a temporal code is sometimes used when simulating spiking network models (Maass et al., 2005; Duarte & Morrison, 2014) but rates are used as well (Litwin-Kumar & Doiron, 2014; Eliasmith, 2013) reflecting the lack of consensus on whether spike times really matter.

3.1.2 Input projection

Orthogonal to the encoding schemes, I also compared two distinct forms of input projections. “Projection” here refers to the connectivity between the input channels and the network. I distinguished between stimulus-specific and stimulus-general projections, see Figure 3.3. *Stimulus-specific projections* connect each input with a subset of neurons that is randomly chosen. This adds a spatial component to the encoding. Some neurons in the network may be exclusively excited by one stimulus, therefore any input-driven activity of such a neuron can be used to identify the stimulus. To investigate how well the encoding schemes can deliver information without this

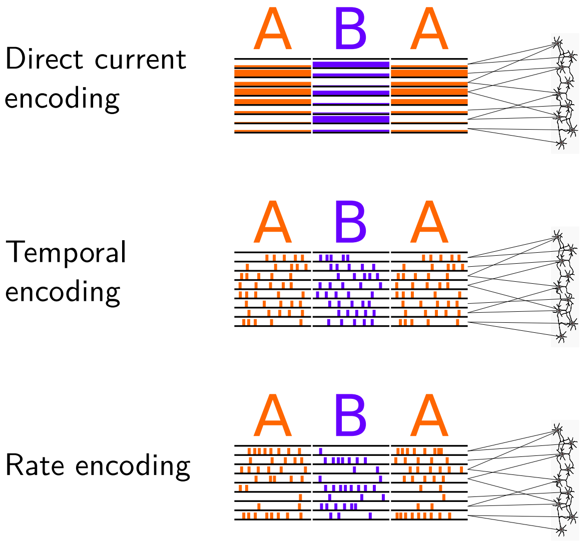


Figure 3.2.: Encoding schemes and input projections. Each panel shows how a series of three input stimuli [A] [B] [A] is encoded in input channels (rows in the spike rasters) where each channel has a fixed connection to a randomly chosen subset of neurons in the network (sketched on the right). The channel activation shows currents for the direct current encoding and spikes for the temporal and rate encoding schemes. Note how currents and spikes differ per channel between stimuli. For the temporal encoding, the spike rate is the same across channels for each stimulus but the timing differs between stimuli. When a stimulus repeats, the precisely timed spike pattern repeats. For the rate encoding the channel rates differ between stimuli. When a stimulus repeats the rate on each channel repeats but not the timing.

spatial component, I tested a second *stimulus-general projection*. In this projection, the same neurons are stimulated by each input and only the current strengths, spike times, or spike rates carry the information necessary to distinguish stimuli from each other. This is implemented by using the same input channels for all stimuli, which are activated differently depending on stimulus identity.

Another projection type that could be tested would activate disjoint sets of neurons for each stimulus. Since the number of stimuli could exceed the network size in the tasks investigated here, such a projection was not practicable.

3.1.3 Benchmark tasks

To evaluate how the different encoding schemes and projections change the processing properties of the NBL model, I tested it on three benchmark tasks that are relevant for the unification module in the Memory, Unification & Control model (Hagoort, 2005). Unification integrates information provided by long-term memory, which

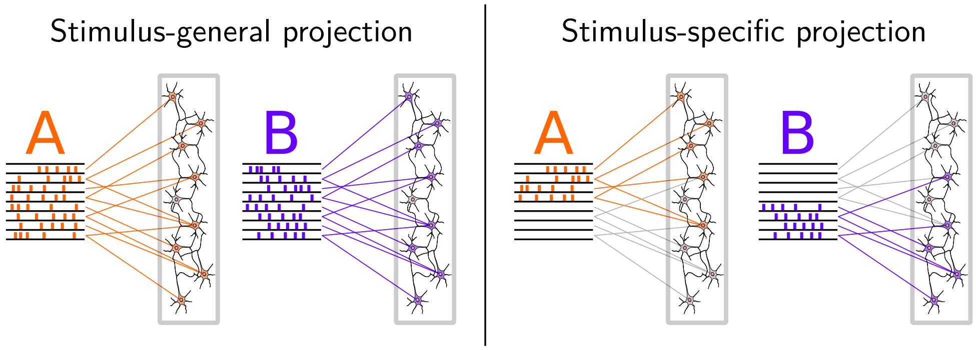


Figure 3.3.: Input projections. The left panel shows how two stimuli [A] and [B] are stimulating the same set of neurons in the network (stimulus-general projection). The right panel shows how [A] and [B] are connected to different subsets of neurons in the network that may partially overlap (stimulus-specific projection). These distinct projections are used for all three encoding schemes.

corresponds to an input sequence of retrieved word representations, into a sentence-level interpretation. To unify a stream of incoming words, they need to be identified correctly. Thus, in the *separability task* I tested how many different input stimuli can be distinguished from each other. Furthermore, to process sequences of inputs, the unification module needs some form of processing memory. In the *memory tasks* I tested how long information remains available in the network. Finally, to form a full sentence interpretation, information from different words needs to be combined. In the *integration task*, I tested a simple form of this where pairs of adjacent stimuli need to be identified after a time delay. This task also requires sensitivity to serial order.

I used randomly generated sets of stimuli that had no internal structure. Thus, the only information a stimulus provides is that it is different from all other stimuli in the set. It is likely that the interface from memory to unification provides richer representations including syntactic and semantic features (Hagoort, 2013), but using unstructured stimuli allows me to avoid specific linguistic assumptions about the feature structure of words.

3.2 Methods

I simulated the NBL model as described in chapter 2. Table 3.1 indicates the parameters I used here that diverge from the general description. The different tasks and encoding schemes I used here are described below because they deviate from the general description chapter 2 and explore more encoding mechanisms than described

Name	Value	Description
τ_w	144 ms	adaptation time constant
τ_{syn}	20 ms	synapse time constant
ρ_c	2%	connection density
f_{tune_n}	7 Hz	network tuning rate
n_{train}	10 000	training set size
n_{test}	2000	testing set size

Table 3.1.: Parameter values used in this chapter

there. The input stimuli were encoded either as a direct current or as a spike pattern delivered over the input channels. The pattern on each channel was generated independently and they were connected to an independently chosen set of neurons in the network.

In the *stimulus-specific projection*, each input had a dedicated set of channels that were only active during this stimulus. For direct currents, this was one channel per stimulus, while in the spike-based encoding schemes I used 5 channels per stimulus. In the *stimulus-general projection*, I used 100 input channels. In section 3.3.1 I varied the number of channels between 10 and 100. All channels were activated by all input stimuli (see Figure 3.3). Thus, only the exact current, rate or spike distribution over the channels carried information to distinguish items and there was no spatial component to input encoding.

The input channels were connected to subsets of the excitatory neurons. The connection probability between a channel and a neuron was 5%, so each channel targeted approximately 40 neurons. The input weights were excitatory and chosen from a uniform distribution between 0 and 1. Stimuli had a length of 50 ms and inputs were presented consecutively without pause. There was an input weight scaling parameter I_{inp} which was tuned such that the evoked spike rate was 2.5 Hz (mean across neurons) when all network-internal connections were set to zero.

With *direct current encoding*, a constant current was injected into the connected neurons and caused them to fire. For the stimulus-specific projection, all currents had the same strength while for the stimulus-general projection, the relative current strengths were drawn from a Gaussian distribution with mean 1 and standard deviation 0.5. The absolute strengths were determined by the scaling parameter I_{inp} and the input weights. In *temporal encoding*, each input pattern was generated before the start of the simulation by a Poisson process with a spike rate of 20 Hz per channel. In section 3.3.1 this rate was varied between 10 and 200 Hz. Whenever a stimulus was repeated in the input, the same spike pattern was used to stimulate the network. In *rate encoding*, spike patterns were generated anew before each stimulus presentation. The pattern was a set of Poisson processes with constant rate per channel for each stimulus. However, the rate of each channel was randomly drawn from a normal distribution. Thus the different stimuli were distinguished by the combination of rates on the input channels. The rate distribution had a mean of 20 Hz and a standard deviation of 10 Hz. In section 3.3.1 the distribution was varied with a mean between 10 and 200 Hz and a standard deviation of 50% of the mean rate (i.e., between 5 and 100 Hz).

Input noise

In section 3.3.4, I tested the robustness of the encoding schemes to noise (all other simulations were noise-free). For spike-based encoding I jittered (i.e., displaced in time) each input spike independently following a Gaussian distribution with the position in the noise-free spike pattern as its mean. The standard deviation σ of the distribution determined the noise level. I used values of $\sigma = 1, 5$ or 10 ms. If a jittered spike reached across the beginning or end of the stimulus, the displacement was inverted in time to maintain a fixed number of spikes per channel. Jittered spikes that went beyond both stimulus boundaries were deleted (less than 1% of all spikes).

Thorpe et al. (2001) discuss a rank coding where the order of spikes encodes information, rather than their precise timing. It is less vulnerable to noise since only the relative rather than the absolute spike times matter (Brette, 2015a). For the noise levels that I applied, each spike swaps its position with 4, 12 and 24 other spikes on average, when $\sigma = 1, 5$ or 10 ms respectively. Thus, for 100 channels, 50 ms stimulus duration, at a 20 Hz channel rate and σ=10 ms, each spike swaps its position with one quarter of all spikes in the stimulus pattern. This represents a substantial amount of noise considering the possibility of a rank-based encoding.

For direct currents, I added white noise to their amplitude. If a current was strong enough to induce regular firing, this created approximately Gaussian jitter in the inter-spike-intervals (Tuckwell, 1988; Gerstner et al., 2014). In simulations, the noise level was chosen such that, in the single neuron case with a fixed current strength, the resulting temporal displacement of spikes in the neuron receiving the input current had approximately the same standard deviation as in the two spike-based encoding schemes.

Importantly, I added noise to the encoding already before the training of the readout, a situation one would also find in cortex where noise is always present.

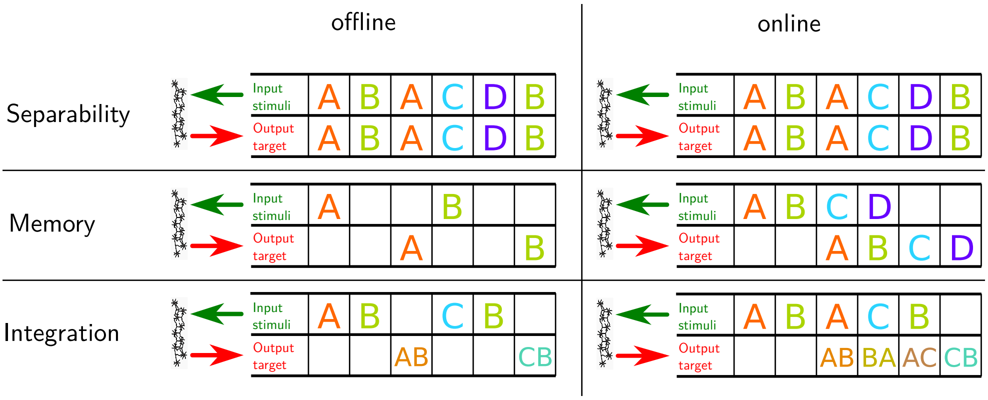


Figure 3.4.: Benchmark tasks. The six panels show example input-output mappings for the three tasks (online and offline version) with a delay of 100 ms (2 stimuli). Input was delivered to the network sequentially and the target output is shown at the appropriate time. The separability task is the zero-delay version of the online/offline memory tasks.

Task description

To test separability, I used an instantaneous identity mapping task where the target output was identical to the input stimulus. To test offline memory, I used a delayed identity mapping task where the target output was delayed by varying numbers of stimuli. During delay, there were filler inputs [x] and there were filler target outputs [x] when there was no other item defined as target output. These correspond to the empty spaces in Figure 3.4 which illustrates the different tasks. Thus, for an input [A] [x] [.] with a delay of 50 ms (duration of one stimulus), the target output was [x] [A] [.] where [.] indicates the end of a sequence. Filler items were represented in the same way as other inputs. Using fillers was preferable to having no input, which tends to shut off network activity, as well as to noise input, which disturbs memory traces in an unpredictable way. In the output, fillers were a separate regression dimension, similar to the other target stimuli, but they were excluded when computing accuracy.

To test online memory, I used an online delayed identity mapping task. Each training and test sequence was a string of inputs followed by filler items [x] for the delay and the end-of-sequence marker. For example, for a delay of 50 ms (duration of one stimulus) this could be [A] [B] [D] [x] [.]. The target output was the same sequence shifted by the duration of the delay, preceded by filler items. In the example, the target would be [x] [A] [B] [D] [.]. In contrast to the variable sequence lengths in the offline tasks, in the online version sequences had a fixed length of 20 stimuli.

To test offline integration, I used a task similar to offline memory but with two stimuli followed by fillers. The target output was coding for a combination of these stimuli. For example, with a delay of 50 ms (duration of one stimulus), the input [A] [B] [x] [.] had the target output [x] [x] [AB] [.], see Figure 3.4 lower left panel. Importantly, also the order of stimuli needed to be identified (i.e., [AB] was different from [BA]). To test online integration, I used a variant of the online memory task with the same continuous sequence of input stimuli. As in the offline integration task, the target output for the online integration task was the combined value of two input items. Here, I also used sequences with a fixed length of 20 stimuli.

In training and testing, an uninterrupted sequence of items with fixed delay was presented. Thus, each end-of-sequence marker was followed by a new sequence of the same type, with items randomly chosen from the stimulus set. All stimuli had the same duration and were randomly generated. The stimulus set size and input-output delay determined the task difficulty. To make the tasks comparable, fillers and end-of-sequence markers were excluded from calculating accuracy and only the accuracy on the output before the end-of-sequence marker was counted in the final score.

3.3 Results

My goal was to evaluate how different encoding schemes influence network processing in terms of separability, memory, and integration. The encodings were crossed with projection type (see Figures 3.2 and 3.3). Since the network parameters remained the same in all simulations, differences in task accuracy were due to different encodings and projections.

3.3.1 Separability and influence of rate and channel number

The first task focused on input separability. The task is an instantaneous identity mapping of input which requires the NBL model to identify an incoming stimulus and output it instantaneously.

Figure 3.5A shows that separability of 1024 different stimuli was 99±0% accurate with the stimulus-specific projection when using direct current input. Stimulus-general projection allowed the identification of 512 different input stimuli with 97±1% accuracy when using temporal encoding. The network consisted of 1000 neurons, so on average two neurons were necessary to encode a stimulus using the general projection. Rate encoding performed worse than the other encodings, and this was more pronounced when using a general projection.

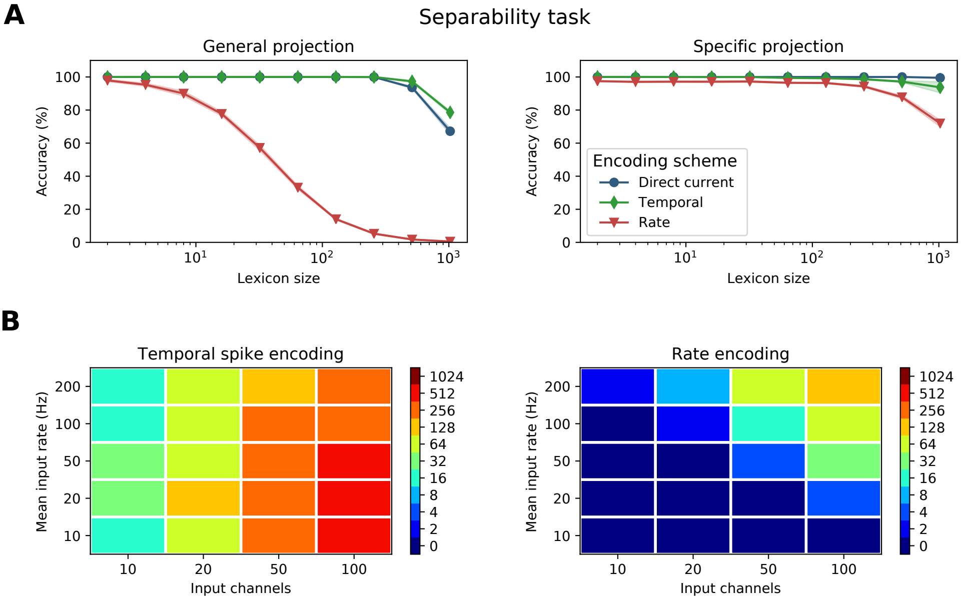


Figure 3.5.: Separability and influence of input rate and channel number. A Accuracy for the instantaneous identity mapping to test separability. The stimulus set sizes are powers of 2 from 2 to 1024, the shading indicates 95% confidence intervals. **B** Influence of input rate and channel number on separability with general projection. The color indicates the largest stimulus set size for which accuracy was above 95%. Set size 0 indicates that accuracy was never above 95%.

To evaluate, whether these separability results depended on model parameters, I varied the number of input channels and the mean spike rate on these channels. Mean spike rate as a parameter only applies to spike-based encodings while input channel number only applies for cases with general projections. Therefore, I only compared spike-based encodings with general projection.

Single-neuron spike rates in mammalian cortex typically lie between 1 and 100 Hz (Markram et al., 2015). There is also evidence for rates of up to 200 Hz in hippocampal ripples (Csicsvari et al., 1999). I used input rates within this range, from 10 to 200 Hz. For the temporal encoding, each channel had the same firing rate for each stimulus, set to the mean firing rate. For the rate encoding, the firing rates varied between channels and stimuli. Those were drawn from a normal distribution around the mean firing rate with a standard deviation of 50% of the mean firing rate. I tested separability for stimulus set sizes between 2 and 1024.

Using linear mixed models, I tested how the separability accuracy was influenced by encoding scheme, channel number and mean spike rates. As dependent variable, I used logit transformed task accuracy and as random effect I used different network subjects. Throughout, I used the maximal random effects structure that still converged (Barr et al., 2013) and p-values were obtained by likelihood-ratio tests.

I found a main effect for the encoding scheme ($\chi^2(1)$=2023, $p<0.001$), the mean spike rate ($\chi^2(1)$=130, $p<0.001$) and channel number ($\chi^2(1)$=380, $p<0.001$). The interactions of the encoding with both the channel number ($\chi^2(1)$=38, $p<0.001$), and the mean spike rate ($\chi^2(1)$=390, $p<0.001$) were significant. I therefore further analyzed the effect of encoding on accuracy for each encoding scheme separately. I observed significant effects for both: For rate encoding, there was a main effect for both channel number ($\chi^2(1)$=430, $p<0.001$) and mean rate ($\chi^2(1)$=400, $p<0.001$). For temporal encoding, there was also a main effect for both channel number ($\chi^2(1)$=260, $p<0.001$) and mean rate ($\chi^2(1)$=12, $p<0.001$). However the latter went in the opposite direction than for the rate encoding: higher accuracy was observed for lower rates.

These results show that the accuracy for both encoding schemes depended on the parameter choices for mean input spike rate and channel number. However, the temporal encoding performed significantly better than the rate encoding, independent of those parameters. In fact, for every parameter combination tested, temporal encoding performed better than rate encoding. For rate encoding, the largest stimulus set that allowed for an accuracy above 95% was 128 for the highest tested firing rate (200 Hz) and the highest tested channel number (100). For temporal encoding, the largest stimulus set that was separable with 95% accuracy was 512, for channel rates up to 50 Hz.

3.3.2 Memory

The memory tasks are based on delayed identity mapping and require the NBL model to identify an incoming stimulus and recall it after a fixed delay period. These tasks come as an offline and an online variant. In the offline memory task, the input stimulus is followed by filler items that do not carry information. The NBL model only needs to retain the information of the stimulus item. In the online memory task, there is an ongoing stream of input stimuli, as it would be in a sequence of words, and the network is required to recall them one by one after a fixed delay, see Figure 3.4.

Figure 3.6 shows the accuracy on the offline and online memory tasks for all combinations of encoding scheme and projection as a function of delay times. In the analysis, I focused on two questions: How does the rate encoding compare to the average of the two other encodings? And how do the temporal encoding and

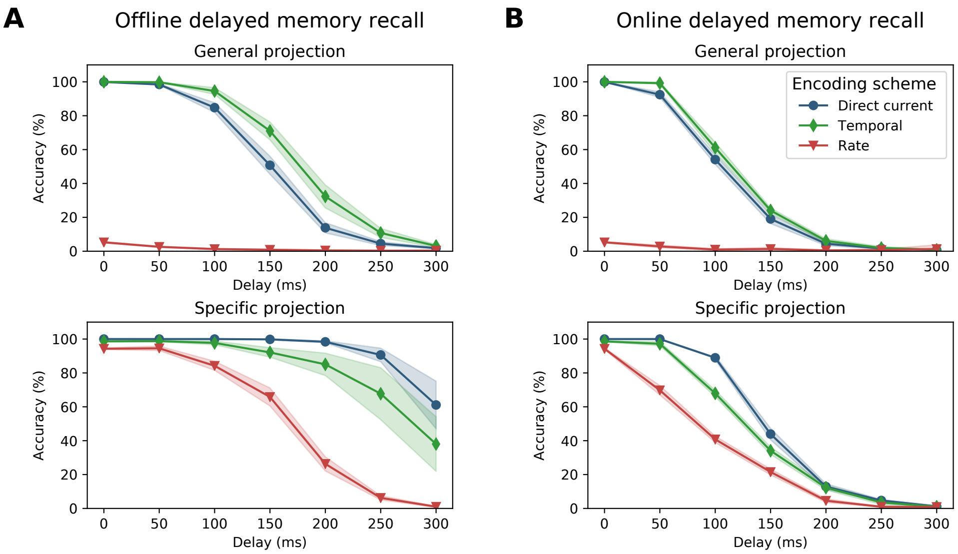


Figure 3.6.: Memory for the different encoding schemes and input projections. Accuracy for the offline and the online memory tasks for different encoding schemes and input projections. The stimulus set size was fixed at 256 items, shading indicates the 95% confidence intervals. The x-axis shows the time delay for recall.

the direct current encoding compare to each other? Furthermore, I investigated how accuracy was influenced by task variant (online or offline) and projection. To answer these questions, I used linear mixed models with logit transformed accuracy as dependent measure. I removed data points where all encodings performed at floor or ceiling. Mixed models had encoding, task variant, and projection as fixed factors and network subject as random effect.

First, I looked at rate encoding compared to the two other encoding schemes across delays. The maximal linear mixed model that converged only had by-subject intercepts. I found a main effect for encoding ($\chi^2(1)=230$, $p<0.001$), projection ($\chi^2(1)=93$, $p<0.001$) and task variant ($\chi^2(1)=49$, $p<0.001$). There was also an interaction of encoding and projection ($\chi^2(1)=23$, $p<0.001$). I therefore further analyzed the effect of encoding on accuracy for each projection separately. I found main effects of the encoding for both projections (both $p<0.001$), as well as an interaction between encoding and task variant in the stimulus-specific projection case ($\chi^2(1)=7$, $p=0.007$) but not for the stimulus-general projection ($p=0.35$). When splitting the data further by task variant, I found simple main effects of encoding ($p<0.001$) for all four combinations of task variant (online/offline) and projection (specific/general).

Overall, these results show that rate encoding performed worse than the other two encoding schemes. For all encodings, accuracy increased with a stimulus-specific projection compared to the general projection. The interaction between task and encoding shows that the difference in accuracy between encoding schemes was sensitive to task variant.

Secondly, I investigated how temporal encoding compared to direct current encoding, across delays. The maximum linear mixed model that converged had by-subject intercepts and by-subject slopes for encoding. I found no main effect for encoding ($p=0.44$) but did find main effects for projection ($\chi^2(1)=55$, $p<0.001$) and task variant ($\chi^2(1)=82$, $p<0.001$). There was also an interaction of encoding and projection ($\chi^2(1)=14$, $p<0.001$). I therefore further analyzed the effect of encoding on accuracy for each projection separately. I only found a main effect of encoding for the stimulus-specific projection. There, the direct current encoding performed better ($\chi^2(1)=9.5$, $p=0.002$) while for the stimulus-general projection, the encoding schemes did not differ significantly ($p=0.11$). There was no interaction between encoding and task variant ($p\geq0.15$) for either projection.

This analysis shows that, overall, there was no difference in accuracy between temporal and direct current encoding. However, for the stimulus-specific projection, the direct current encoding had a higher accuracy.

3.3.3 Integration

To test a simple form of information integration, I use the delayed adjacent item integration tasks and test how well the combination of two subsequent stimuli can be retrieved after a fixed delay time. The readout needs to retrieve the identity of the pair including the presentation order after a given delay time after the presentation of the first item of the pair. I use this task again in an online and an offline variant (see Figure 3.4).

Figure 3.7 shows the accuracy in the online and offline integration tasks, for all combinations of encoding scheme and projection type as a function of delay times.

As before, I focused on two questions in the analysis: How did the rate encoding compare to the average of the two other encoding schemes? And how did the temporal encoding and the direct current encoding compare to each other? Furthermore, I investigated how accuracy was influenced by task variant (online or offline) and projection. I used linear mixed models as described above.

First, I analyzed rate encoding compared to the two other encodings across delays. The maximal linear mixed model that converged only had by-subject inter-

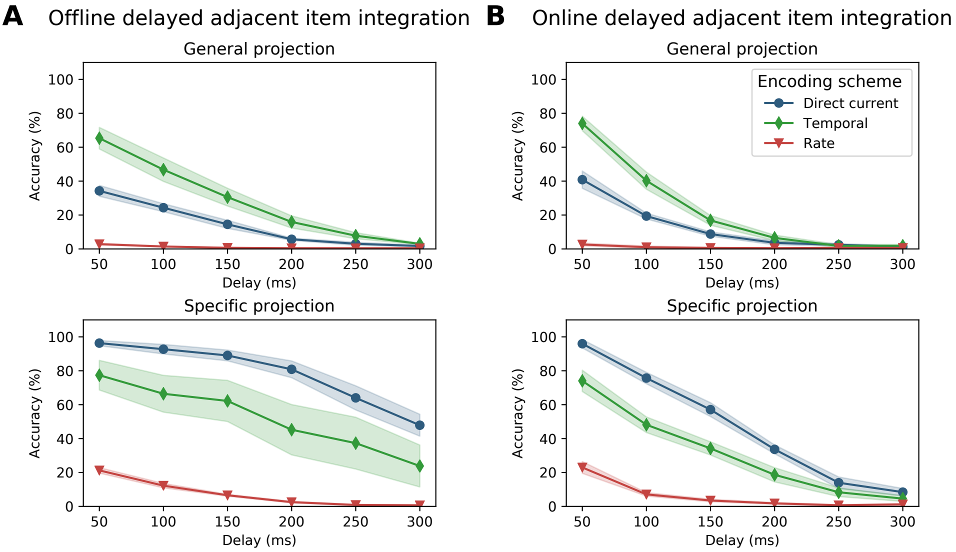


Figure 3.7.: Integration accuracy for the different encoding schemes and input projections. Accuracy for a fixed stimulus set size of 16, shading indicates the 95% confidence interval. Time delay for integration shown on the x-axis.

cepts. I found a main effect for encoding ($\chi^2(1)=470$, $p<0.001$), projection ($\chi^2(1)=140$, $p<0.001$) and task variant ($\chi^2(1)=8$, $p<0.001$). There was also an interaction of encoding and projection ($\chi^2(1)=12$, $p<0.001$). I therefore further analyzed the effect of encoding on accuracy for each projection separately. I found main effects of the encoding for both projections (both $p<0.001$). I found an interaction between encoding and task variant in the stimulus-specific projection case ($\chi^2(1)=11$, $p<0.001$) but not for the stimulus-general projection ($p=0.64$). When splitting the data further by task variant, I found simple main effects of encoding ($p<0.001$) for all four combinations of task variant (online/offline) and projection (specific/general).

Secondly, I looked at how temporal encoding compares to rate encoding across delays. The maximal linear model that converged had by-subject intercepts and by-subject slopes for encoding. I found no main effect for encoding ($p=0.058$) but I did find main effects for projection ($\chi^2(1)=180$, $p<0.001$) and task variant ($\chi^2(1)=27$, $p<0.001$). There was also an interaction of encoding and projection ($\chi^2(1)=57$, $p<0.001$). I therefore further analyzed the effect of encoding on accuracy for each projection separately. I found main effects of encoding for both projections. For the stimulus-specific projection, the direct current encoding performed better ($\chi^2(1)=36$, $p<0.001$)

while for the stimulus-general projection, the temporal encoding performed better ($\chi^2(1)=17$, $p<0.001$). There was no interaction between encoding and task variant ($p \geq 0.18$) for either projection.

These results show a similar pattern as for the memory tasks. Overall, rate encoding performed worse than the other two encoding schemes and the accuracy for all encodings improved with a stimulus-specific projection. The interaction between task and encoding shows that the difference in accuracy between encoding schemes was sensitive to task variant. Furthermore, there was a difference in accuracy between temporal and direct current encoding, but only when testing both projections separately. For the stimulus-general projection, the temporal encoding performed better while for the specific projection, the direct current encoding performed better.

3.3.4 Noise tolerance

One argument against a temporal neural code is its potential vulnerability to noise (Shadlen & Newsome, 1998; London et al., 2010). To assess this issue, I investigated the effect of noise on the temporal encoding and compare it to the noise tolerance of the other encoding schemes. Before presentation to the network, the temporal position of each input spike is jittered independently, following a Gaussian distribution around the original position. The standard deviation σ of this distribution determines the noise level. For the direct current encoding, I added white noise to the input current that jittered spike positions in the receiving neuron compared to the noise-free case. The standard deviation of the jitter then corresponded to the noise level in the spike-based encoding schemes (see methods for a detailed description).

I found that for a stimulus set size of 256 and a delay of 50 ms, the accuracy of the temporal encoding in the online memory task dropped by less than 5% for σ=5 ms and less than 15% for σ=10 ms, depending on the projection (see Figure 3.8A). The spike frequency in each input channel was 20 Hz. Therefore, σ=10 ms corresponds to a noise distribution with a standard deviation of 20% of the average inter-spike-interval (see Figure 3.8B for an example).

To analyze these results, I again used linear mixed models with logit transformed accuracy as dependent measure. The models had noise level, encoding, and projection as fixed factors and network subject as random effect. I tested the three-way interaction between noise level, encoding, and projection with by-subject intercepts. Here, encoding had the three levels: direct current, temporal and rate encoding. I found main effects for all three factors (all $p<0.001$) but also a three-way interaction

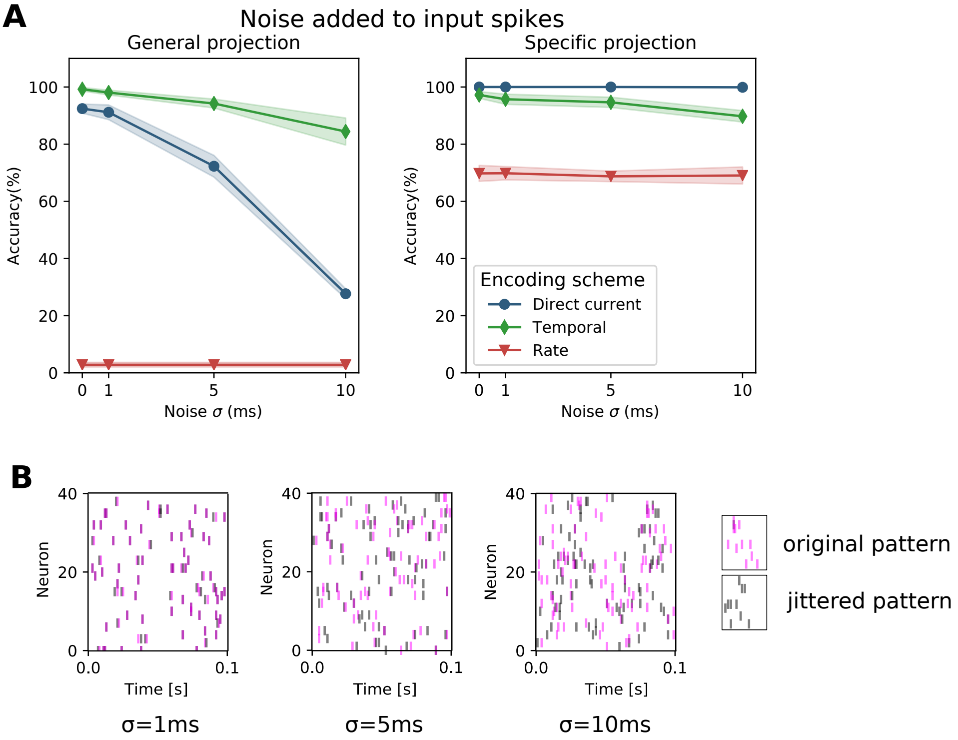


Figure 3.8.: Noise tolerance of the encoding schemes. **A** Effect of noise level σ on different encoding schemes and different stimulus projections during online memory task with a stimulus set size of 256 and a delay of 50ms. Shading indicates the 95% confidence intervals. **B** Example patterns for different levels of noise. For σ=1ms, original pattern and jittered pattern largely overlap (dark magenta marks) while σ=10 ms generates significantly different spike patterns. The time shown is 100 ms which is the presentation time of two stimuli. The example is taken from a stimulus-general projection where the same input channels encode different stimuli. In total there were 100 of such channels, 40 are shown here.

($\chi^2(2)$=120, p<0.001) and all three two-way interactions were significant as well (all p<0.001).

To further analyze the data, I split it by projection. Instead of looking at all encoding schemes at once, I split the data to answer the same questions I addressed in earlier sections: How does the rate encoding compare to the average of the two other encoding schemes? And how do the temporal encoding and the direct current encoding compare to each other?

First, I looked at rate encoding compared to the two other encodings, separately for stimulus-general and stimulus-specific projection. The maximal linear mixed models that converged in both cases had by-subject intercepts and, for the specific projection only, by-subject slopes for noise level. For the specific projection, I found a main effect for the encoding ($\chi^2(1)$=140, p<0.001, here rate compared to the other encodings) and for the noise level ($\chi^2(1)$=3.9, p=0.05) but no interaction between the two (p=0.2). For the general projection, I found main effects for both encoding ($\chi^2(1)$=190, p<0.001) and noise level ($\chi^2(1)$=53, p<0.001) and also the interaction between them ($\chi^2(1)$=39, p<0.001).

In these results, the interactions between noise and encoding are of particular interest. Since there was no significant interaction for the stimulus-specific projection, it means that rate encoding and the average of the other two encodings are equally robust to noise. For the stimulus-general projection, there was an interaction between encoding and noise, indicating that the other two encodings were less robust against noise than rate encoding. However, one has to consider the low accuracy of rate encoding in the noise-free case compared to the other two encoding schemes, i.e., accuracy was zero already without noise and could not decline any further.

Secondly, I looked at how temporal encoding compared to rate encoding, split by projection. The maximal linear models that converged had only by-subject intercepts. For the stimulus-specific projection, I found a main effect for the encoding ($\chi^2(1)$=150, p<0.001, here temporal compared to direct current encoding) and the noise level ($\chi^2(1)$=27, p<0.001) and an interaction between both ($\chi^2(1)$=25, p<0.001). For the stimulus-general projection, I found main effects for both encoding ($\chi^2(1)$=37, p<0.001) and noise level ($\chi^2(1)$=160, p<0.001) and also an interaction between them ($\chi^2(1)$=13, p<0.001). However, the direction of the interaction was dependent on the projection: for the specific projection, the direct current was more robust than temporal encoding while for the general projection the temporal encoding was more robust than the direct current encoding.

3.4 Discussion

In this chapter, I investigated whether the processing characteristics of a simulated spiking network depended on the chosen input encoding. I tested three encoding schemes, direct currents, precisely timed spike patterns, and spike rates, each with either stimulus-general or stimulus-specific input projection (see Figures 3.2 and 3.3). I found that the choice of encoding scheme had a significant impact on the performance of the NBL model in terms of separability, delayed recall, and integration. Overall, rate encoding performed substantially worse than direct current and temporal encoding. Comparing the latter two with each other, I found a difference in performance depending on the projection. When using a stimulus-specific projection the direct current encoding performed best, while with a stimulus-general projection, the temporal encoding performed better. I also found that the specific projection yielded higher accuracy compared to the general projection.

These results agree with theoretical arguments which suggest that temporal encoding transmits information more efficiently than rate encoding (Thorpe et al., 2001). They also agree with the results of Duarte et al. (2018) who investigated how different encoding schemes affect separability. The two encoding schemes compared by Duarte et al. (2018) correspond to the direct current encoding with stimulus-specific projection on the one hand, and the temporal encoding with stimulus-general projection on the other hand. In agreement with my results, they found that the former yielded better performance. However, since I tested both projections in combination with both encoding schemes, my results indicate that this difference could mainly be due to the difference in projections. Duarte et al. (2018) speculated that the performance difference between the two encoding schemes was reduced for more complex tasks. In agreement with that, I found that the performance difference due to projection was reduced in the online task variants compared to the offline variants.

I found that all but one combination of encoding scheme and projection could discriminate 256 stimuli in the separability task (accuracy of >90%) with a network of 1000 neurons. The exception was the rate encoding combined with a stimulus-general projection which only yielded an accuracy of 5±1%. However, for larger stimulus sets there were bigger differences in accuracy between combinations of encoding and projection: with the general projection, the temporal encoding performed best and with the specific projection, the direct current encoding performed best (the latter distinguished 1024 stimuli with 99% accuracy). This shows that the number of stimuli that are separable is on the order of the number of neurons in the network. These results speak against mean field or neural mass models that reduce the number of degrees of freedom in a simulation from one-per-neuron to one-per-population (Deco et al., 2008). Such models describe the activity of a group of neurons with a probability function over the neuron population, a mean firing rate, which implies the use of a rate code. My results suggest that this would reduce the separability properties of the network significantly.

One concern with a temporal code has been that it is potentially sensitive to noise which is considered problematic given the observed variability of neuron responses to repeated stimulus presentations (Shadlen & Newsome, 1998; London et al., 2010; see Brette, 2015a for discussion). The reasoning is that, if every single spike and its temporal position carries information, a shift in time or the omission of a spike changes the information that is encoded. To address this issue, I conducted simulations where the spikes of the temporal encoding were jittered by Gaussian noise and found that this encoding only showed a small to moderate performance decrease even for high noise levels, suggesting that the temporal encoding used here is robust to noise. Even when the performance was reduced by noise, in all tested instances temporal encoding performed better than rate encoding. These results can be explained given the higher information density in a temporal code. In other words, using a temporal code, a fixed number of spikes can encode more information than the same number of spikes can using a rate code (Thorpe et al., 2001). This increased coding capacity can be used by the NBL model to connect multiple spike patterns, which are noisy instantiations of each other, to the same information. Instead of a temporal code where the exact timing of a spike matters, this results in a more robust code where a time window of an individual spike carries the information. Since the noise in my simulations is already present in the training phase (as it would be in the neurobiological system), the width of such a time window can be adapted to the present noise level and thus ensure reliable information encoding. Importantly, such a coding scheme is *not* a form of rate coding, it is a spike-based temporal code that nevertheless does not rely on absolute spike times (Brette, 2015a).

The main results are based on a fixed set of parameters for the neuron model, network connectivity and the encoding schemes. These were chosen to be within a neurobiological range, but it is conceivable that the particular parameter choices determine the performance of each encoding scheme, or which one is the best. Similarly, the task parameters could be chosen differently which may lead to different results. Two choices are the number of input channels, especially for the stimulus-general projection, and the spike rates on these channels. To see how these choices affected accuracy, I tested how these parameters influenced separability. I found that the difference between the two spike-based encoding schemes got smaller with higher spike rates. However, within biologically plausible firing rates below 200 Hz (Markram et al., 2015; Csicsvari et al., 1999), the results did not change qualitatively. Interestingly, the highest separability for the temporal encoding was found right at the most commonly observed post-stimulus firing rates in mammalian cortex, at ∼20 to 50 Hz (Markram et al., 2015).

The model did not include synaptic dynamics which would allow the network to adapt to input. Spike timing dependent plasticity (STDP) regulates synaptic strength depending on the precise timing of pre-synaptic and post-synaptic spikes (Markram et al., 1997). STDP is a suitable mechanism for neural networks to develop sensitivity to an arbitrary set of spike patterns (Masquelier et al., 2008). Therefore, if STDP would be added to the simulations, it is likely to increase the sensitivity of the network to the temporal encoding scheme. Synaptic dynamics that run on longer time scales, for example, realized through synaptic tagging (Frey & Morris, 1997), would provide additional memory and improve the network's capability to solve tasks with longer delays. How synaptic plasticity mechanisms, connectivity patterns, and neuronal properties interact on a network level is an important open research question. Results in this study suggest that input encoding should be included in the investigation of these interactions to understand how all of them together contribute to the computational properties of brain circuits.

The encoding schemes I tested differ in terms of their biological plausibility. Direct current does not represent a way cortical neurons typically communicate and can be seen as a noise-free benchmark for neuronal communication. I found that temporal encoding performs similar or better, especially with the stimulus-general projection and in the online integration task, which was the most complex task tested here. Temporal encoding is also preferable since direct current encoding comes with conceptual limitations: the output of one network – spikes – cannot be directly used to excite another higher-level network. When using spike-based encoding schemes, it becomes possible to directly connect different network modules and use appropriate adaptation rules to develop the output of specific spike patterns (Legenstein et al., 2005; Pfister & Gerstner, 2006; Memmesheimer et al., 2014). Such a setup may be useful for modular simulations of cortical computations (Eliasmith, 2013) including models of language production (Chang, 2002). With a progressing understanding of the details of neural computation, it is likely that such modular models will become more common in neuroscientific and linguistic research.

From a psycholinguistic point of view, I investigated the nature of a neurobiological interface between the memory and unification component in the MUC model (Hagoort, 2005). The input channels would be the neuronal fibers in, for instance, the arcuate fasciculus, delivering information from posterior memory regions of the brain to prefrontal unification regions. My findings suggest that the neural code for transmitting information on these fiber structures could rely on spatio-temporal spike patterns. Given the inefficiency of rate encoding I found, the interface between memory and unification, which needs to transmit complex linguistic information (Hagoort, 2013), is less likely to use a rate code.

Given the distributed nature of the neurobiological language network and especially the memory regions in the temporal cortex, there is likely to be some level of stimulus-specific projection to the prefrontal unification component. However, considering individual fiber tracts, such as the arcuate fasciculus, my results suggest that a stimulus-general projection is more suitable. This would mean that individual axons are not used solely for one word or piece of information, but that the activity pattern transmitted over a fiber bundle collectively encodes the information. Given the coding efficiency and the number of stimuli I could distinguish in the separability task, it seems plausible that the information transmitted is either based on word identity or on a more complex data structure encoding linguistic information.

The results presented here suggests that it is important to systematically investigate how encoding schemes modulate the processing characteristics of neural networks in the effort to understand the neural code. Furthermore, the interactions between different encoding schemes and task complexity suggest that the investigation of simulated neural networks should use complex, cognitively plausible, tasks.

4

Neuronal processing memory for language comprehension

4.1 Introduction

Language is context-dependent at all levels of processing. Letters and phonemes form syllables, syllables form words, words form phrases and clauses which then combine into sentences, pieces of discourse, et cetera. To combine these units into larger structures, information needs to be integrated over different time scales, from milliseconds to seconds and beyond (Hasson et al., 2015). This hierarchy of time scales is not exclusive to language processing, but important also in other cognitive domains (Bernacchia et al., 2011; Murray et al., 2014; Gjorgjieva et al., 2016). From a neurobiological point of view, cognitive processes are computed in cortical networks by neurons and their interconnecting synapses. One element of neuronal communication is the action potential. It travels between neurons within a hundredth of a second, has a fixed duration of around one millisecond, and a generic shape that is independent of the amount of synaptic drive (Luo, 2015). Thus, neural signaling is one or two orders of magnitude faster than the time scales of cognitive processes. It is an outstanding scientific question of how a system based on short-lived action potentials can provide memory on time scales that are relevant to cognition and behavior (Chaudhuri & Fiete, 2016).

In this chapter, I examine some memory principles in the NBL model. I simulate language processing in networks of spiking neurons in real physical time, and investigate what mechanisms can give rise to a processing memory that is suitable for sentence comprehension. Specifically, I study how networks with different memory characteristics resolve semantic relations between words in sentences. I will argue that processing memory for language can be provided by experience-dependent changes in neuronal excitability.

4.1.1 Memory as persistent activity

The long-term storage of information is often associated with changes in synaptic conductances (Koch, 1999; Takeuchi et al., 2014), although this view has recently been challenged (Gallistel & King, 2011). Short-term memory, in turn, is often conceptualized in terms of persistently active neural assemblies. On this account, assemblies are activated by an incoming stimulus and subsequently show an elevated level of activity that persists beyond stimulus offset (Amit & Mongillo, 2003; Durstewitz et al., 2000). Thus information is represented and maintained through sustained activity. This view of memory on short time scales is supported by studies that have used delayed match-to-sample tasks in behavioral decision making. In these tasks, monkeys are exposed to stimuli on a screen (e.g., a colored dot) and have to move their eyes to a memorized screen location after a brief delay period. During the experiment, neural activity is recorded from, for instance, the prefrontal cortex. Several studies have shown that there was elevated, stimulus-selective activity during the delay which was interpreted as a neural correlate of short-term memory (Funahashi et al., 1989; Fuster & Alexander, 1971; Kubota & Niki, 1971). These findings have made persistent activity the leading candidate model of memory on short time scales (Curtis & D'Esposito, 2003). This type of short-term memory works as a passive storage device that maintains information until the information is read (and potentially deleted) by a readout mechanism, similar to register memory in computers (Jacob et al., 2010). However, more recent studies have found evidence suggesting that activity may not persist under certain task demands. In these studies, monkeys were trained to perform another task during the delay, with increased demand for attention, and found that neural activity did not reliably persist during the delay period (Watanabe & Funahashi, 2014). Instead, there was a ramp-up reactivation immediately before the saccade which was interpreted as a shift of attention, and suggest that elevated firing may not be necessary for short-term memory (Eichenbaum & Cohen, 2004). Thus, at present the role of persistent activity in short-term memory maintenance remains unclear (see Stokes (2015) and Sreenivasan and D'Esposito (2019) for recent reviews).

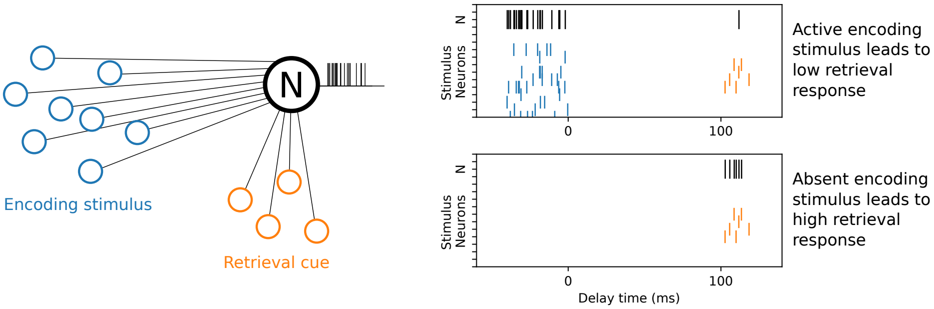


Figure 4.1.: Spike rate adaptation causes history-dependent neuronal responses. Illustration of how the AdEx neuron *N* (black output spikes) can retain information encoded by a stimulus (blue spikes) due to spike rate adaptation. After a period without input (here 100 ms), a retrieval cue (orange spikes) triggers a response by the AdEx neuron that indicates if there was an earlier stimulus. If there was one, the neuron reacts weakly to the retrieval cue (upper right panel) while otherwise the neuron reacts more strongly to the retrieval cue (lower right panel). This shows that spike rate adaptation cause history-dependent responses through an activity-silent memory mechanism.

4.1.2 Activity-silent memory

An alternative account of short-term memory has focused on processes of synaptic adaptation as a storage device. This was implemented computationally in networks with short-term synaptic facilitation (STF) (Markram et al., 1998; Mongillo et al., 2008). In these simulations, information is encoded by activating two different subsets of neurons. The different input stimuli create distinct patterns of functional connectivity because of short-term synaptic facilitation. After a delay, a generic retrieval cue is sent into the network, causing different patterns of spiking activity for the different inputs. From these spike patterns, the identity of the memorized stimulus could be decoded reliably in the absence of persistent activity during delay. This work shows that networks with short-term synaptic facilitation can memorize stimuli and recall them explicitly using a retrieval cue.

Another theory of memory on short time scales has recently been proposed in Fitz et al. (2020) where information was stored and maintained in short-lived neuronal adaptation (Figure 2.4) rather than synaptic changes. On this account, spiking activity triggers adaptive currents which temporarily hyperpolarize the neuronal membrane and this can be viewed as writing information into memory. This is consistent with experimentally observed bursts of neuronal activity during the encoding stage (Lundqvist et al., 2016). Because these currents are coupled to the membrane, information is also continuously retrieved from memory which also changes the functional state of the neuron. What these two accounts – neuronal and synaptic short-term memory – have in common is that they do not require persistent activity to maintain information, and they use dynamic variables for information storage that are distinct from the membrane potential. In Figure 4.1 I illustrate this proposal where information is stored in adaptive conductances. I simulated a single AdEx neuron N and excited it with a strong input stimulus (blue spike pattern). Neuronal adaptation is clearly visible in the spike response pattern (black output spikes). Then, the neuron did not receive further input for 100 ms and therefore stopped spiking. After a delay, I excited the neuron with a retrieval cue (orange spikes) and because of the earlier input, the neuron responded weakly with a single spike to the retrieval cue. In contrast, when there was no initial stimulus (lower right panel), the activity of the neuron triggered by the retrieval cue was much stronger. This shows that information can be stored using spike rate adaptation and this information can be read out by a retrieval cue. The simulation also demonstrates that neuronal memory is context-dependent. The retrieval cue can be seen as information that is processed differently depending on contextual information (presence or absence of the blue stimulus). This property of memory on short time scales is highly relevant for language processing which requires the constant integration of words with their linguistic context in a processing memory.

Synaptic and neuronal memory differ from persistent activity accounts in that information storage is provided by physiological processes other than spiking activity. These processes supply dynamic variables that determine the current state of the network and change on slower time scales than the evolving membrane state. This perspective on memory is summarize in Figure 4.2. Action potentials are short-lived ($\sim$1 ms) and the membrane potential reacts quickly to new input and integrates information over a time window on the order of tens of milliseconds. In contrast, neuronal and synaptic adaptation responds more slowly to input and typically integrates information over longer time scales. For example, spike rate adaptation (SRA) decreases a neuron's sensitivity given high levels of input and acts over hundreds of milliseconds (Connors & Gutnick, 1990). Synapses elicit a synaptic current that excites or inhibits the post-synaptic neuron when they are stimulated. These currents can be long-lasting and have time constants of up to 1000 ms (Gerstner et al., 2014). These types of slower neuronal and synaptic processes can in principle serve as information storage since information encoded in those variables is accessible for several hundreds of milliseconds (Mongillo et al., 2008; Fitz et al., 2020). Because the neural and synaptic changes induced by the adaptation mechanisms are different

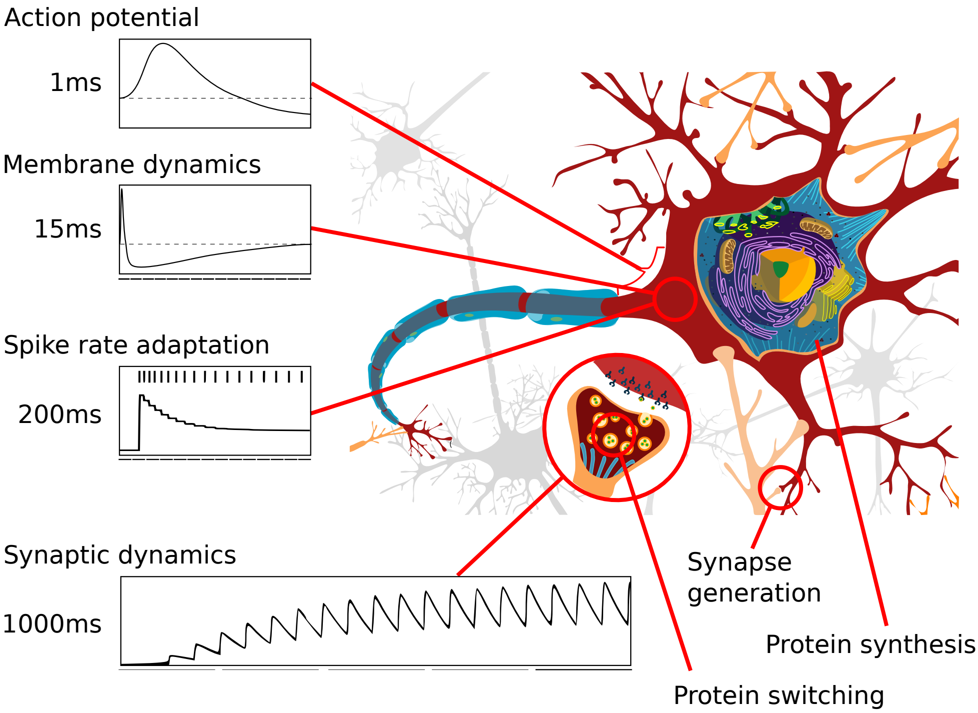


Figure 4.2.: Time scales of dynamic variables in physiology. Biophysical processes in neurons and synapses are active on different time scales spanning several orders of magnitude. These range from short-lived action potentials and fast membrane dynamics to neuronal spike rate adaptation and short-term synaptic plasticity. Processes such as protein synthesis, switching, and synapse generation can act on longer time scales. All of these physiological processes can potentially support memory.

from changes in membrane voltage and spiking activity, they have also been termed *activity-silent* mechanisms (Barak & Tsodyks, 2014; Stokes, 2015).

4.1.3 Constraints on memory for language

The persistent activity model for memory has been developed in the context of delayed response tasks. In these tasks, a small number of items have to be remembered and recalled explicitly after a delay following a retrieval cue. Memory requirements for language processing, however, differ from this paradigm. During language comprehension, a rapid stream of input words needs to be combined to generate a sentence-level interpretation. In terms of the number of memorized items, already simple sentences quickly exceed the limits of explicit recall in short-term memory (Miller, 1956). Furthermore, linguistic information is not limited to word identity but carry lexical meaning as well as morphological and syntactic features. These cues need to be integrated over variable temporal windows and at different sentence positions in order to arrive at a coherent interpretation. Another important aspect of many languages is that word order matters for their joint interpretation. For instance, sentences such as *"The child bit the dog"* and *"The dog bit the child"* give rise to different sentence-level interpretations but they use the same collection of words. This means that information is integrated taking the precedence relation of words into account. It is unclear how this can be realized in simple models of persistent activity memory since only the occurrence of individual words is memorized while their precedence order is lost (Ganguli et al., 2008). In order to perform fast sentence comprehension, it has been argued that short-term memory needs to be accessible in a content-addressable manner through retrieval cues generated by the word input (Lewis et al., 2006). Memorized information constantly influences the state of the network and therefore also processing. This means that explicit retrieval cues are not required for the integration of information over time. Rather, integration happens continuously in a state-dependent manner. Similarly, different types of information (e.g., morphological, syntactic or semantic) are constantly integrated into the dynamical state of the network in order to generate an interpretation of utterances as the processing outcome. I refer to it as state-dependent unification (Hagoort, 2005; Petersson & Hagoort, 2012; Frank & Fitz, 2016; Fitz et al., 2020).

4.1.4 This study

Here, I investigate the nature of memory for language processing using the NBL model. I stimulated the network with sequential language input and tested how different parameters influenced its memory characteristics. As a readout target, I used the semantic role of the word that was currently presented to the network. I tested hypotheses derived from the persistent activity memory model and whether network connectivity influenced the performance in a manner that is consistent with these hypotheses. Furthermore, I tested hypotheses derived from the activity-silent memory account and whether the time scales of neuronal adaptation and synaptic currents affected the performance correspondingly.

4.2 Methods

I simulated the NBL model as described in chapter 2. Table 4.1 indicates the parameters I used here that diverge from the general description.

Variable	**Value**	**Description**
τ_w	from 10 ms to 1 s	neuronal adaptation time constant
τ_{syn}	from 10 to 500 ms	synaptic current time constant
ρ_c	from 1% to 10%	connection density
f_{tune_n}	from 3 Hz to 10 Hz	network target spike rate
n_{train}	20 000 words ≈ 2200 sentences	training set size
n_{test}	5000 words ≈ 550 sentences	test set size
a	$4.0\,\mathrm{n}\Omega^{-1}$ $(0\,\Omega^{-1})$	neuronal sub-threshold adaptation conductance
b	80.5 pA (0 pA)	spike-triggered increment of neuronal adaptive current

Table 4.1.: Parameter values used in this chapter. When a and b were set to 0, spike rate adaptation was disabled.

When investigating the influence of connection density ρ and target spike rates f_{tune_n} in section 4.3.1, I used τ_{syn}=20 ms and τ_w=144 ms, following the proposed parameter values in Brette and Gerstner (2005). When testing the influence of spike rate adaptation and its time constant τ_w in section 4.3.2, I used recurrent networks with ρ_c=5% and f_{tune_n}=5 Hz while keeping τ_{syn}=20 ms. I used these same parameters together with variable values for τ_{syn}, in section 4.3.3. Finally, when comparing performance to reference models in section 4.3.4, I used a recurrent network with parameters ρ_c=5%, f_{tune_n}=5 Hz, τ_{syn}=500 ms and τ_w=200 ms.

Feed-forward connectivity

To investigate the influence of recurrence, I compared recurrent networks (generated as in chapter 2) with feed-forward networks. These were designed to have the same number of neurons and synapses with the same weight distribution as the recurrent networks but without recurrent connections. To achieve this, I generated a connectivity matrix as for recurrent networks but with twice the connection probability $\rho_{c_{ff}} = 2 \cdot \rho_c$. Then, I deleted the lower triangle of the matrix with randomly permuted columns and inverted the permutation afterwards. This permutation was used to ensure that inhibitory and excitatory synapses were deleted with equal probability.

4.2.1 Semantic role assignment task

As network input, I used the language described in section 2.3. Sentences from the language were delivered to the network sequentially. Each of the words had a target semantic role. I trained a readout from the states of the NBL model to assign a semantic role to each word instantaneously, as it was being processed. By design, this role assignment did not only depend on the identity of the current word but also on the preceding sentence context. I refer to the readout performance on this task across all words in the tested sentences, excluding the end-of-sentence markers, as *overall performance*. For many words, especially early in a sentence, *semantic role assignment* was an ambiguous task since the disambiguating information to resolve it correctly was not yet available. However, there was always sufficient information at the last word of a sentence to assign the semantic role unambiguously (i.e., there were no globally ambiguous sentences in the language). Furthermore, since the semantic roles of the sentence-final words required processing memory that could retain information from the beginning of the sentence, they required the longest memory span. Therefore, I used the readout performance on the *final word* to evaluate the memory properties of the network. The readouts for overall and final word performance were both estimated from the entire network state collection, recorded across all input words. All performances reported in this chapter were kappa corrected (see mathematical glossary, section 2.4) with P_{MAX} set to 100% for final words. The mean performance of a random classifier across 10 seeds was $P_{rand} =$ 17.24% for the overall and P_{rand} =12.14% for the final words.

4.3 Results

4.3.1 Memory through recurrence

The persistent activity model assumes that a population of neurons, once activated by a particular stimulus, uses recurrent connections within the population to remain active, even when the stimulus has been removed. In this way, information is maintained by an elevated level of activity. First, I tested whether processing memory in static, randomly connected networks of spiking neurons was provided by recurrent connectivity. If recurrence was required for memory, (i) recurrent networks should outperform feed-forward networks with the same connection density. Furthermore, (ii) memory should increase with higher connection density since the number of cycles grows substantially with the number of connections in a network of constant

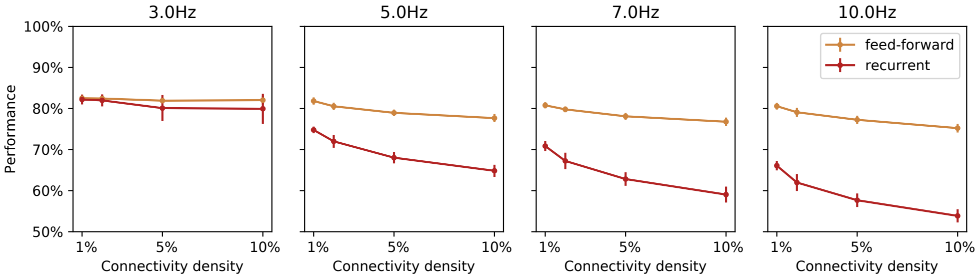


Figure 4.3.: Effect of connectivity parameters on performance. Semantic role assignment on the sentence-final word by the NBL model for different combinations of mean network spike rates (shown on top of plots) and connection density for feed-forward and recurrent networks. Error bars show 95% confidence intervals.

size. I tested this by comparing networks with different densities between 1 and 10%. And finally, (iii) stronger synapses should enhance memory since they support stronger recurrent feedback connections. I regulated the synaptic strength in the network with a global scaling parameter that tuned the average neuronal spike rate to 3, 5 or 10 Hz while keeping input strength and the number of recurrent synapses constant. Thus, higher spiking activity in the network was due to stronger recurrent connections. The results of these comparisons are shown in Figure 4.3.

I applied linear mixed effect models to test how logit-transformed performance was influenced by the structural network parameters. Categorical predictors were effect-coded. In general, I used the maximal random effects structure that still converged (Barr et al., 2013). In this section, this was only the network subject as intercept. Throughout, p-values were obtained by likelihood-ratio tests. Using a mixed model with the three-way interaction between network recurrence (feed-forward/recurrent), connection density, and mean spike rate, I found main effects for all three factors ($p<0.001$). In particular, the main effect for recurrence was such that feed-forward networks outperformed recurrent ones ($\chi^2(1)=300$, $p<0.001$). There was no three-way interaction ($p=0.54$) but two-way interactions between all three pairs of factors ($p<0.01$ each). To further investigate the data, I split it according to recurrence. I found main effects for connection density, and mean spike-rate for both recurrent and feed-forward networks ($p<0.001$ each). Importantly, the effects were such that both higher connection density and higher spike rates caused lower performance.

These results can now be evaluated in light of the hypotheses derived from the persistent activity model. Hypothesis (i) predicted reduced performance with feed-forward networks compared to recurrent ones. I found that the opposite was the

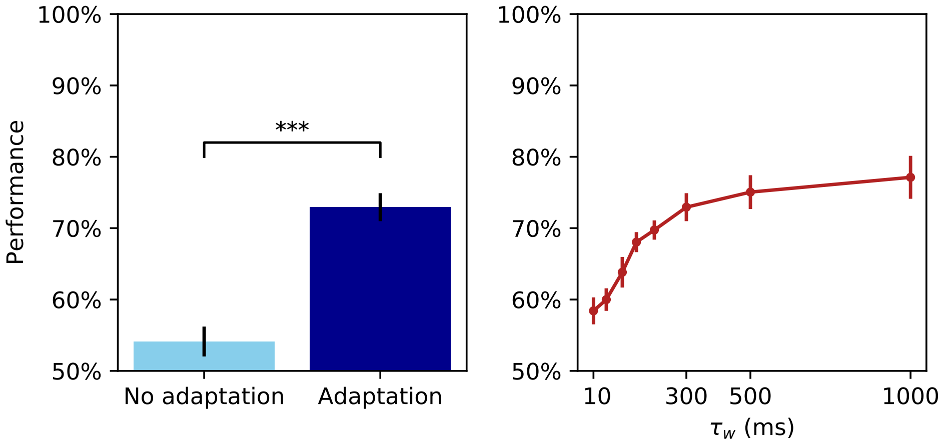


Figure 4.4.: Influence of spike rate adaptation time constants on performance. The left panel shows the performance of the NBL model with and without adaptation (τ_w=300 ms). The right panel shows how performance with adaptation enabled depends on its time constant. Error bars show 95% confidence intervals.

case in that feed-forward networks performed significantly better than recurrent ones. Hypothesis (ii) predicted better performance with higher connection density. I again found that the opposite was the case and lower connection densities delivered better performance. And finally, hypothesis (iii) predicted better performance with higher mean network spike rate, but here also, the opposite was the case. Lower spike rates lead to significantly better performance than higher rates. Thus, I could not confirm any of the predictions derived from the persistent activity model. This suggests that recurrent feedback, as postulated by persistent activity accounts, cannot fully explain the processing memory characteristics in the NBL model.

4.3.2 Memory through spike rate adaption

Next, I investigated the alternative proposal for processing memory that relies on processes of intra-neuronal adaptation rather than connectivity. The neuron model I used to simulate networks incorporated spike rate adaptation that influence the neuronal activity on time scales longer than 100 ms. If spike rate adaptation supports memory, performance should be low if there is no adaptation and it should be higher for long time constants compared to shorter ones. To test this, I compared networks with and without adaptation (figure 4.4, left panel) and investigated how the length of the adaptation time constant τ_w influenced performance (figure 4.4, right panel).

To evaluate the influence of adaptation statistically, I applied a linear mixed model with adaptation (enabled or disabled) as fixed factor. Random effects included the network subject as intercept and by-subject slopes for the adaptation. Otherwise, the statistical modeling was done as described above. I found a main effect of adaptation ($\chi^2(1)$=21, p$<$0.001) showing that without adaptation, performance was significantly worse than with adaptation. I then tested for variation in the time constant τ_w when adaptation was enabled. Using a mixed model with τ_w as fixed factor and intercepts for network subjects as the only random effect, I found a main effect of length ($\chi^2(1)$=71, p$<$0.001) where longer time constants lead to better performance. These results are in line with the predictions of the activity-silent memory model and they indicate that neuronal spike rate adaptation supports processing memory in the simulated networks.

4.3.3 Memory through synaptic currents

To broaden the perspective on possible mechanisms for processing memory, I also investigated whether synaptic processes could be utilized. I focused on synaptic currents that can have a non-instantaneous effect on post-synaptic neurons. When a synapse is activated by a spike, the synaptic conductance is increased, generating a current that excites or inhibits the post-synaptic neuron. The shape, amplitude and duration of this current differs between synapse types but synaptic transmission can happen on much longer time scales than the spike itself (Gerstner et al., 2014) and therefore provide a memory of the preceding stimulus history. I tested this by simulating the NBL model with different synaptic time constants τ_{syn} from 10 to 500 ms. To also investigate the interaction with neuronal adaptation, I tested each of these cases with τ_w ranging from 10 to 500 ms (see Figure 4.5).

To analyze the results statistically, I used a linear mixed model with τ_{syn} and τ_w as fixed factors. Random effects included the network subject as intercept and by-subject slopes for τ_{syn}. Otherwise, the statistical modeling was done as described above. I found a main effect for both τ_{syn} ($\chi^2(1)$=52, p$<$0.001) and τ_w ($\chi^2(1)$=120, p$<$0.001). In both cases, longer time constants improved performance. In addition, I found an interaction effect between synaptic and neuronal time constants ($\chi^2(1)$=59, p$<$0.001). These results are shown in figure 4.5. In the previous section, I had used τ_{syn}=20 ms. For this value, performance improved from short to long adaptation time constants. Hence, for short synaptic time constants, processing memory can be enhanced by longer neuronal time constants. However, for long synaptic τ_{syn}, it is less clear how spike rate adaptation contributes to memory. For τ_{syn}=500 ms the best

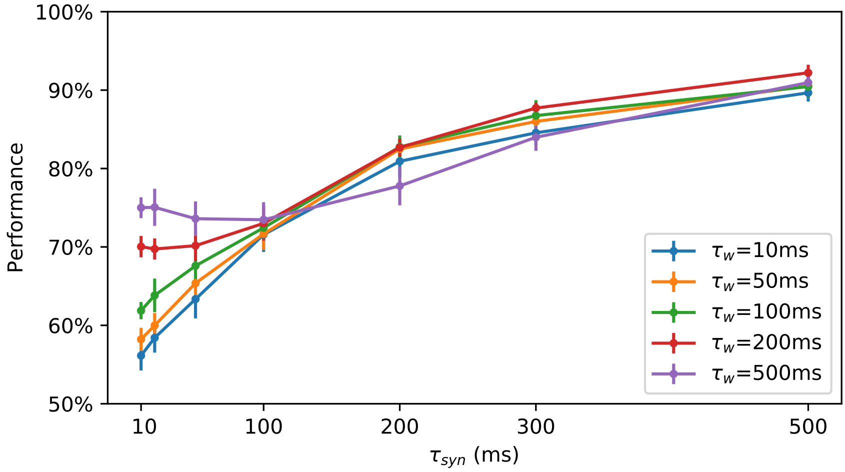


Figure 4.5.: Long synaptic current time constants enhance memory. Performance depends on both synaptic current and neuronal adaptation time constants. Error bars show 95% confidence intervals.

performing models had a shorter τ_w=200 ms. Using a linear mixed model with data limited to τ_{syn}=500 ms and only τ_w as fixed factor, I found no main effect (p=0.085). Thus, for long τ_{syn} an increase in τ_w was not improving performance.

4.3.4 Comparison of NBL model performance with reference models

To put the semantic role assignment task into perspective, I compared the NBL model to two other reference models that did not have a neural network architecture. In the first model, the linear readout used in the NBL model was directly applied to the sequence of input words. Since the readout does not possess memory on its own, I call it the *memory free readout model*. The difference in performance compared to the NBL model illustrates the contribution of the spiking network itself. As the second reference model, I stored all N-grams from all sentences of the training set together with the target semantic role for the last word of each N-gram. I then used a back-off approach to assign the semantic role for the largest known contextual chunk in memory (see section 2.2.3 for a detailed description). In case of ambiguous context (multiple semantic roles are possible), the back-off N-gram model assigned the role that occurred most frequently in this context. This model had perfect memory of the sentence context and stored all N-gram-related statistical information from the

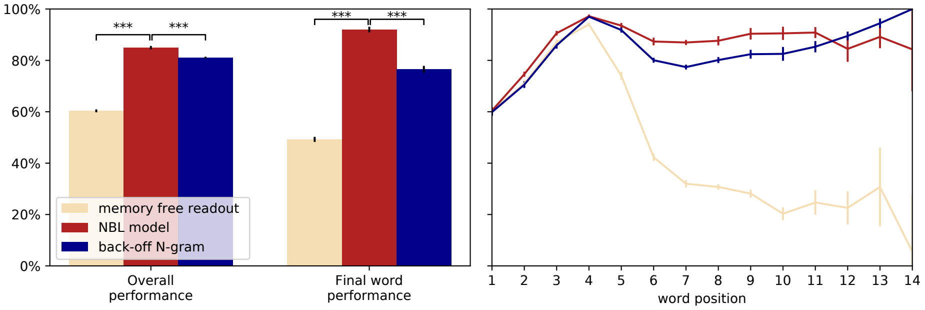


Figure 4.6.: Comparison of NBL model with reference models. The left panel shows average role assignment performance for all words and on the sentence-final word. The right panel shows performance by word position in a sentence. Error bars show 95% confidence intervals. Note that due to different sentence lengths, the sentence-final word can occur anywhere between the 2nd and 14th position. Only a few sentences were longer than eleven words which causes small sample sizes and larger confidence intervals for late positions.

sentences of the training set. The results from these model comparisons are shown in Figure 4.6. As before, performance was measured on the final word of each sentence. At this point, all sentences were non-ambiguous and the longest processing memory was required to assign semantic roles correctly. In addition, I investigated overall performance, that is, the percentage of correct role assignments on all words of the sentences in the testing set.

The memory free readout assigned the correct role on the sentence-final word in 49±1% of the cases while the NBL model performed at 92±1% ($\chi^2(1)=87$, $p<0.001$). The overall performance was 60±1% and 85±1%, respectively ($\chi^2(1)=100$, $p<0.001$). The statistical analysis was performed analogous to previous sections, with logit-transformed performance (overall or final word) as dependent variable and model type as fixed factor. The random effects included only intercepts for network subjects. The difference in overall performance between the back-off N-gram model and the NBL model was small but still significant ($\chi^2(1)=46$, $p<0.001$) with 81±0% compared to the 85±1% of the NBL model. Performance on the sentence-final word was also higher for the NBL model with 92±1% compared to 77±1% for the back-off N-gram model ($\chi^2(1)=58$, $p<0.001$). Thus, the NBL model outperformed both reference models by a significant margin.

4.4 Discussion

In this chapter, I used the NBL model to evaluate different theories about the neurobiological basis of processing memory where memory span was tested during a sentence comprehension task. I found that in the NBL model, processing memory is provided by activity-silent processes such as spike rate adaptation and synaptic currents with long time constants. At the same time, I could not find evidence that recurrent connectivity alone could provide better processing memory than feed-forward connectivity, a central tenet of the persistent activity account. In general, recurrent connectivity was detrimental to semantic role assignment relative feed-forward connectivity with activity-silent processes in that it decreased processing memory somewhat.

I also found that the NBL model outperformed two reference models. The fact that neither of these models performed at ceiling shows that the semantic role assignment task was non-trivial and could not be solved using word identity or input statistics. The NBL model performed significantly better than the back-off-N-gram model which had perfect memory of lexical context. This shows that it can generalize semantic roles from known lexical contexts to new contexts which the back-off-N-gram model was unable to do. The NBL model also performed better than the memory free readout which shows that the spiking network contributed significantly to processing memory.

Overall, the findings in this chapter are similar to Fitz et al. (2020) but my simulations used a different neuron model and method of input encoding. Instead of leaky integrate-and-fire neurons with adaptive currents I used the AdEx neuron and a spatio-temporal spike pattern as encoding scheme, in contrast to direct current input. Despite these differences, the results found here are qualitatively the same as in Fitz et al. (2020). However, they generally found higher performance than I report. This aligns well with the difference in encoding strategies used. For instance, I showed in chapter 3, that a stimulus-specific projection with direct currents, as used in Fitz et al. (2020), results in improved memory and separability characteristics compared to the spike pattern-based encoding I used here. Nevertheless, the similarity of both sets of results supports their validity independent of the implementation details. In particular, it supports the notion of an activity-silent processing memory as a viable alternative to the persistent activity view when tested in the language domain.

The simulations did not include background activity and used noise-free input. Furthermore, due to the network reset at the beginning of each sentence, there was no influence of previous sentences on network activity. The only sources of variability were the input sentences themselves – each presented sentence was unique – the random network connectivity and input spike patterns which were also randomly generated. How much this noise-free environment contributed to high performance remains to be tested. However, I showed in chapter 3 that with the encoding method I used here, memory was also robust to noise. Spiking network simulations in the computational neuroscience literature typically model random background activity due to constant stimulation from external sources found in cortex (Litwin-Kumar & Doiron, 2014; Duarte & Morrison, 2014; Zenke et al., 2015; Brette et al., 2007). However, the random nature of such observed activity has also been contested (Brette, 2015a; Rieke & Warland, 1999). It is possible that this activity is not background noise but provides contextual information to the processing network. In fact, on single presentation of each input pattern, my encoding scheme is indistinguishable from noise (see section 2.2.2). Thus, an expanded version of the model tested here, with larger input patterns containing additional information, could be an appropriate representation of a cortical network that receives background input.

Within the NBL model, I investigated sources of processing memory for randomly connected spiking networks without synaptic plasticity. It is an open question how this extends to more complex neuronal models and biological learning. We know that biological neurons display a range of different dynamic processes that continuously shape the neuron's behavior. However, given that the two temporally extended processes I investigated here could both serve as a source of processing memory, it is plausible to assume that other dynamical processes with long time constants can similarly support memory. In biological neurons, different neuronal processes build a spectrum over different time scales. Synaptic currents have time constants ranging from 6 ms to more than 1000 ms (Gerstner et al., 2014). Synaptic dynamics such as long-term potentiation and depression have time constants of tens of seconds and longer (Zenke et al., 2015). Other processes such as synaptic tagging (K. C. Martin & Kosik, 2002), protein synthesis (Fallon & Taylor, 2013) and synaptic growth and pruning (Chechik et al., 1998) act on even longer time scales. These processes can create a hierarchy of time scales supporting memory.

The activity-silent and the persistent activity view are not mutually exclusive. Both could be active in parallel and complement each other to provide processing memory. However, since I did not find a benefit of recurrent connectivity in randomly connected networks, persistent activity based memory would require a particular cortical connectivity structure. It is conceivable that such connectivity is generated by incremental synaptic changes or developed during evolution. We know that each neuron class has a specific connectivity profile to other neuron classes (Markram et al., 2015; Izhikevich & Edelman, 2008). With such connectivity, biological networks could be suitable to store information through persistent activity by refreshing information through cyclic chains of connections. What speaks against this is a consideration of time scales. Information transfer between neurons is fast. For one neuron to excite another can take between 1 ms and 20 ms (Markram et al., 2015; Izhikevich & Edelman, 2008). Thus, to maintain information for the same period of time as a neuron-internal dynamic process with a time constant of 200 to 1000 ms, up to 1000 consecutive neuronal spike interactions are required. Since neurons are rarely excited by single spikes, the number of involved spike interactions would be even higher. This makes it more resource intense to store information in persistent activity rather than activity-silent processes – an option biology also provides.

In this study, I did not find a functional role of recurrent connectivity for memory. However, recurrent connectivity is a prevalent feature in cortex and therefore likely to serve an important role. What might explain this discrepancy is that recurrence might be particularly important for learning and/or more advanced processing demands beyond the current task. Biologically more plausible learning might be dependent on local feedback provided by recurrent connectivity (Whittington & Bogacz, 2019). Furthermore, it is plausible that recurrence is required to solve tasks that involve a more complex nonlinear recombination of information than what I tested.

To process a sentence, it has been argued that the language system needs to temporarily store information and flexibly retrieve it when needed (Gallistel & King, 2011; Graves et al., 2016). One way to conceptualize this is in terms of memory registers that are accessed by independent read and write operations. The activity-silent memory mechanism discussed here presents an alternative view of memory for language processing. It relies on continuous reading and writing between different dynamical processes (see also Petersson et al., 2008). The processes that act on longer time scales serve as activity-silent memory registers that locally encode information through spike generation. At the same time, these processes continuously influence other dynamical processes acting on shorter time scales, such as the membrane state and spiking activity. This can be viewed as a read-from-memory operation (see Figure 4.7). Therefore, information in the membrane potential is locally encoded in memory variables through spike generation. Those variables then again locally affect the future state of the membrane potential. Conceptually, this read-write memory is described by the coupled differential equations that characterize the neuronal

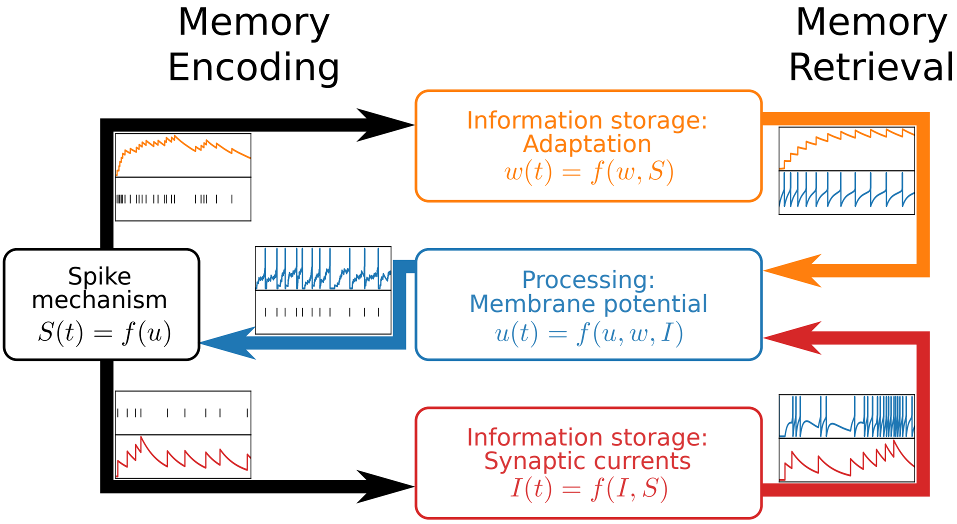


Figure 4.7.: Illustration of memory encoding and retrieval cycles in the NBL model. Information is processed by the membrane potential which can utilize both neuronal adaptation and synaptic currents to store information. A change in the membrane potential can trigger spikes which write information into both dynamic variables, w and I for storage. Those variables in turn continuously influence the membrane potential u. Thus, information in the memory variables is continuously read out and influences processing. Spike, voltage and current traces at each of the arrows illustrate how the variable that is sending information can change the state of the coupled variable. The processes on the left show writing/encoding through spikes while the processes on the right are reading/retrieval operations.

dynamics. Continuous reading and writing might explain why the language system can rapidly process sentences in an online, incremental manner.

In conclusion, my results show that the NBL model relies on a neuronal mechanism for activity-silent processing memory. The way that dynamical variables interact here to implement reading and writing arguably agrees more with the requirements for sentence processing than a random access model. Such a neurocentric model of processing memory supports a shift from abstract models of memory and sentence processing towards models informed by neurobiological constraints.

5

The NBL model and human sentence processing

5.1 Introduction

One purpose of a cognitive model of sentence processing is to explain aspects of human sentence processing in mechanistic terms. It aims to show similar strengths, weaknesses and processing effects as humans do when they process sentences (Demberg & Keller, 2019). This distinguishes cognitive models from natural language models and algorithms in computational linguistics. These are typically focused on applications, such as automatic speech recognition (Varga & Moore, 1990), question answering (Ravichandran & Hovy, 2002), or statistical machine translation (Brown et al., 1993), and try to eliminate shortcomings, regardless of whether these reflect human language use. In contrast, the assumptions underlying a cognitive model need to be tested by comparing model behavior to experimentally observed human behavior. In this chapter, I will connect the NBL model to psycholinguistics and evaluate it against the background of experimental results from human sentence processing.

The NBL model has not been explicitly designed to reproduce human-like sentence processing behavior. Rather, it is built to reflect important aspects of cortical dynamics in a simulated network of spiking neurons. However, since the model aims to capture critical features of the neurobiological infrastructure that performs sentence processing in humans, one should expect some similarity in processing behavior. In fact, if a neurobiological model would faithfully reflect neuronal activity in the language network, with adequate models of neurons and their interaction, such a model would exhibit the full range of human sentence processing ability. However, simulations at this level of detail are not yet feasible because, on the one hand, not all relevant neurobiological details are understood and, on the other hand, it would consume computational resources that are currently not available. Thus, I will investigate which aspects of human sentence processing behavior already emerge given the level of detail simulated in the present NBL model. Unlike for a cognitive model, I will not investigate cognitive mechanisms and modeling assumptions. Rather, I will test the consequences of the neurophysiological data that informed the NBL model on the example of sentence processing. This means I investigate the consequences of the biological assumptions that went into the NBL model to the sentence processing behavior and what strengths and weaknesses readily emerge that are similar to human sentence processing. Demberg and Keller (2019) identified a set of well-established psycholinguistic effects that capture key characteristics of syntactic processing in humans and argued that cognitive models of sentence processing should reflect these properties. Below, I introduce a number of well established experimental findings that are partially overlapping with the items in Demberg and Keller (2019). If the NBL model is a useful first approximation of the neural infrastructure for sentence processing, it should be able to reproduce at least some of these effects.

Sentence processing is fast, incremental and predictive

Humans process sentences incrementally and start constructing a sentence-level interpretation while the utterance is still unfolding. This has been shown both in behavioral (Marslen-Wilson, 1975; Marslen-Wilson & Tyler, 1980; Morris, 1994) and electrophysiological studies (Kutas & Hillyard, 1980; van Petten, 1993; Kutas & Federmeier, 2011). Each input word triggers an update of the partial interpretation constructed thus far, and information is integrated as soon as it becomes available. Processing is fast and reevaluation can occur several times during a sentence (Marslen-Wilson, 1975). At the same time, during incremental processing predictions are made about upcoming linguistic material. This has been shown in terms of reaction times which become faster when a comprehender is confronted with predictable compared to unpredictable input (Arnon & Snider, 2010; Traxler & Foss, 2000; see Kuperberg & Jaeger, 2016 for a comprehensive review). These findings indicate that predictions are made when possible and that these predictions can facilitate the processing of new input. Kuperberg and Jaeger (2016) also analyze various notions of prediction used in the literature and identify *prediction in the minimal sense* as “the context influences the state of the language processing system before the bottom-up input is observed.” I use this notion of prediction in this chapter.

There is an ongoing discussion concerning the benefits of prediction and how ubiquitous it is in language comprehension. Often, there are thousands of possible sentence continuations, and therefore some have doubted that prediction is a worthwhile investment of resources (van Petten & Luka, 2012; Jackendoff, 2002). Others have argued that, although prediction is observed in language processing, it may not be fundamental to comprehension. For example, Huettig (2015) suggests that prediction might only be relevant for learning, for some parts of language processing or under particular conditions, for example for learning, or in conditions that are particularly challenging for a listener. Another controversial issue concerns the nature of predicted material. One perspective is that only a single prediction is made at any given time, for example about the syntactic structure of a sentence. If this prediction is disconfirmed, a full reanalysis is necessary. However, recent evidence suggests that prediction might be a graded phenomenon, depending on how predictive the context is. For instance, EEG studies have found that the size of the N400 effect, which is evoked by unexpected words, depends on the level of surprisal of the incoming word in relation to its linguistic context (DeLong et al., 2005; Frank et al., 2015; Nieuwland et al., 2018). This suggests that the language system may predict several words in parallel and that these predictions reflect a probability distribution over plausible continuations of a given sentence (Fitz & Chang, 2019). However, it remains a contentious issue whether graded predictions are made in parallel or in a serial fashion where only one prediction is made at any given time (van Gompel et al., 2005; Kuperberg & Jaeger, 2016).

The garden path effect illustrates both fast, incremental processing and prediction. Sentences eliciting garden path effects have been a cornerstone of sentence processing research since Bever (1970). They are semantically and syntactically legal sentences that are constructed to elicit a strong expectation in the reader while the sentence is still unfolding (e.g., *The experienced soldiers warned about the dangers conducted the midnight raid*). Later parts of the sentence violate this expectation and require a reanalysis of the sentence.

Multiple sources of information are used in sentence interpretation

The information used to interpret sentences and predict upcoming linguistic material does not come from a single source but instead originates from several different linguistic features (Altmann & Mirković, 2009). For example, predictions are made based on lexical semantics and sentence context (van Petten, 1993) as well as world knowledge (Hagoort et al., 2004). There is an ongoing debate whether prediction is also based on lower-level linguistic features, such as phonology, morphology and syntactic cues (DeLong et al., 2005; Altmann & Mirković, 2009) which has recently been questioned (Nieuwland et al., 2018).

Similarly, the predictions that are made based on these cues affect multiple levels of processing. For example, predictions can facilitate syntactic (E. Gibson & Wu, 2013) and semantic processing (Kuperberg et al., 2011). Furthermore, predictions are made about different grain sizes, ranging from specific words via semantics (Chambers et al., 2002) to more coarse-grained semantic categories (Altmann & Kamide, 1999). Predictions can also facilitate processing on higher levels such as event structure and sentence structure to determine *who did what to whom* (Altmann & Kamide, 1999; Kuperberg & Jaeger, 2016).

There does not seem to be a "level specific" prediction effect, for example that phonological cues would only facilitate phonological processing (Jackendoff, 2007). Rather, cues from any level can inform predictions and therefore facilitate other levels of language processing. For example, contextual information, such as discourse and world knowledge, can facilitate the processing of syntactic structure (E. Gibson & Wu, 2013).

Systematicity

Generalization is another important aspect of sentence processing. Hadley (1994) uses "systematicity" as a technical term that refers to the ability to generalize from a small and sparse training set to a larger set of unfamiliar sentences and constructions; I will adopt this notion here.

Fodor and Pylyshyn (1988) criticized connectionist models by arguing that they can not deal with the combinatorial syntax and semantics necessary for systematicity. Hadley (1994) operationalized this criticism and distinguished weak and strong systematicity (among others): A model exhibits *weak systematicity* if it can successfully process sentences it has not seen before while it has seen all components (such as nouns and verbs) of a sentence individually in each of their syntactic position. A model exhibits *strong systematicity* if it (a) exhibits weak systematicity and (b) it can additionally process sentences with components in syntactic position which the model has not encountered before. This means the set of sentences with which the model is trained excludes, for example, one particular noun from one particular syntactic position. A model with strong systematicity would then be able to correctly process that noun in that syntactic position. An extension, which may be implied by Hadley's notion of strong systematicity, is that the model can infer properties of

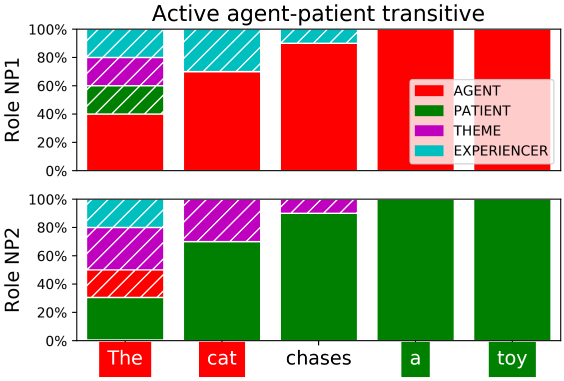


Figure 5.1.: Example of a full-sentence semantic role assignment plot. Illustration of semantic role assignment to two noun phrases (NP) over the course of a sentence (not based on simulated data). The upper and lower panels show the frequency of the respective roles for the first noun phrase *the cat* and the second noun phrase *a toy* for the agent-patient transitive construction at each position. The full bars mark the correct semantic roles while the hatched ones indicate incorrect assignments. The sentence at the bottom is a representative of the construction average displayed in the bar plots. In this example, the initial role assignment is changing due to accumulating evidence from noun and verb identity.

novel words from context. This would correspond to the human capacity to infer syntactic and semantic properties of unfamiliar (pseudo)words from context (Berko, 1958). Hadley's analysis sparked a series of papers investigating the systematicity of connectionist models, aiming to demonstrate strong systematicity at least under certain conditions (Christiansen & Chater, 1994; Hadley et al., 2001; Frank, 2006; Fitz & Chang, 2009). How this notion of systematicity relates to the human capacity to generalize in language is still unresolved (Calvo & Symons, 2014).

5.1.1 Full-sentence semantic role assignment

To compare the NBL model to human sentence processing, I calibrated the readout on input sentences from the English-like language described in section 2.3 and assessed the model's ability to assign semantic roles. Semantic roles provide a shallow sentence-level interpretation (Jackendoff, 1992). They differ from syntactic dependency relations in that the roles are insensitive to syntactic alternations and describe the event structure related to an action. To observe how the interpretation of a sentence is changing while the sentence unfolds, I read out the semantic roles of all noun phrases after the presentation of each word (see Figure 5.1). This *full-sentence semantic role labeling* differs from the task used in chapter 4 where a semantic role was only assigned to the currently presented word. Here, already in the early stages of each sentence, the NBL model provides an assessment of the semantic structure of the entire sentence. Thus, it needs to predict the roles of the upcoming noun phrases and determine "who did what to whom" using the available information. To do this, the model needs to constantly integrate the information from new words and update its semantic role assignments for the whole sentence. Furthermore, previous information relevant for semantic role assignment at a later stage needs to be retained by the network itself as there is no external memory. I implemented this task using parallel readouts, one dedicated for each noun phrase of the sentence. These readouts are trained independently of the other readouts. As previously, each readout had access to the activity of the whole network in the NBL model.

5.1.2 This study

I used this predictive, incremental task to test how well the model could interpret sentences in terms of semantic roles and resolve temporary ambiguities over time. I also tested whether the model could use multiple semantic and syntactic constraints to arrive at the correct sentence interpretation. Finally, I tested the systematicity of the model, i.e., whether it could infer semantic roles of familiar words in a novel context and whether it could infer semantic roles of completely novel words from context.

5.2 Methods

I simulated the NBL model as described in chapter 2, informed by the insights from chapters 3 and 4. Table 5.1 displays the model parameters that I used here that diverge from the general description.

Variable	**Value**	**Description**
τ_w	200 ms	neuronal adaptation time constant
τ_{syn}	500 ms	synaptic current time constant
ρ_c	5%	network connection density
f_{tune_n}	5 Hz	network target spike rate
n_{train}	160 000 words $\approx$ 18 000 sentences	training set size
n_{test}	20 000 words $\approx$ 2200 sentences	testing set size

Table 5.1.: Parameter values used in this chapter.

The training set was larger than in previous chapters. This was necessary because semantic role assignment to whole sentences was more difficult than previous tasks. Note, however, that the number of training items was still less than 0.000005% of all possible sentences generated by the language.

5.2.1 Readout calibration

Since the tested sentences could have up to three noun phrases, I used three independent, parallel readouts that were trained on the same network state collection but had different target outputs. Each readout mapped onto the semantic role of one noun phrase in the input sentences. For sentences with fewer than three noun phrases, the target for the redundant roles was NONE. For example, in an intransitive sentence like *The cat sleeps* with only one noun phrase, the second and third semantic role targets were NP2:NONE, NP3:NONE. Where indicated, readout performances were kappa corrected (see mathematical glossary, section 2.4).

5.2.2 Input language

The input language (section 2.3) was modified slightly. First, the distinction between past participle *-ed*^{par} and the past tense marker *-ed* was removed such that passives and past tense verb forms were indistinguishable (both used *-ed*). This allowed me to investigate sentences with temporary ambiguity in section 5.3.3. Secondly, to test systematicity in the NBL model, I excluded sentences with selected words from the training set and only allowed them in the testing set. This concerned three nouns from the LIVING category and three nouns from the OBJECT category. For one noun of both categories, each sentence that contained either was removed from the training set. These two nouns were used to test systematicity on *novel words*. For the other four nouns, sentences containing them were only removed if they belonged to specific constructions to test systematicity on *familiar words in novel contexts*; see Table 5.2 for examples. Sentences with one noun from LIVING were removed if they were passive agent-patient transitives (with either constraining or unconstraining beginning or with an instrumental prepositional phrase). Sentences with one noun from OBJECT were removed if they were active agent-patient transitives. Similarly, sentences with one noun from LIVING were removed if they were active theme-experiencer constructions and sentences with one noun from OBJECT were removed if they were passive theme-experiencer constructions. In each case, the sentences that were removed had the novel word in the second noun phrase. This ensured that there

Noun category	Example noun	Constructions removed	Example sentence
LIVING	man	*all*	**removed:** The man chases the orange. **removed:** The orange hurts the man.
OBJECT	ball	*all*	**removed:** The boy chases the ball. **removed:** The ball hurts the boy.
LIVING	woman	agent-patient transitive passive	**removed:** The orange is chased by the woman. **included:** The woman chases the orange.
OBJECT	toy	agent-patient transitive active	**removed:** The boy chases the toy. **included:** The toy is chased by the boy.
LIVING	girl	theme-experiencer transitive active	**removed:** The toy hurts the girl. **included:** The girl is hurt by the toy.
OBJECT	apple	theme-experiencer transitive passive	**removed:** The boy is hurt by the apple. **included:** The apple hurts the boy.

Table 5.2.: Examples of nouns that were omitted in the training set and the corresponding constructions to test systematicity. Constructions with any of six selected nouns were removed from the training set. The Table shows the respective noun category with an example noun and the name of the construction that was removed if it contained the noun. On the right are examples of sentences that would be removed or included in the training set.

was sufficient contextual information available for the model to assign the correct semantic role. For the constructions investigated, the semantic role of the second noun phrase was always fully determined by the preceding context.

5.2.3 Explanation of stacked bar plots

To visualize how full-sentence semantic role assignment changed over time, I use stacked bar plots. These plots show the percentages of roles assigned to all sentences of the same construction type in the test set (y-axis) for each sentence position (x-axis). The construction type is indicated on top of each plot. Percentages were averaged over ten model subjects with different random connectivity, training and test sets, and different input spike patterns. To improve readability, error bars were omitted. The color of each bar indicates the assigned role according to the legend, and colors were kept consistent across plots. The solid colors mark the correct roles, the hatched ones mark incorrect assignments.

Since parallel readouts were used, there are three semantic roles assigned at each position – one for each noun phrase in the sentence. For Figure 5.6, I only selected the readout for the relevant noun phrase, but the complete plots for all noun phrases are included in the appendix.

Figures 5.3 and 5.5 show the same role assignments, however, the x-axis does not mark sentence positions, but the positions of a specific word category within sentences of the construction type. The example word on the axis label is one representative of this category. Figures 5.4 and 5.6 mark the sentence positions on the x-axis. However, since the language was designed to reflect aspects of the variability in human language, the same position in different sentences of a construction type can be occupied by words of different classes. For example, for the double object dative in Figure 5.4A, one of the shortest possible sentences is *Girl -s give man -s book -s* with 7 words. Here, I consider suffixes such as *-s* and *-ing* as separate words. The noun positions are 1, 4 and 6 and the verb position is 3. One of the longest possible sentences from this construction is *The young girl -s were give -ing the old man -s the big book -s* with 15 words. The noun positions are 3, 10 and 14 and the verb position is 6. Thus, information available at a given position can differ between sentences of the same construction type. Averaging for each position has the effect that plotted role assignments change gradually because the disambiguating cue can occur at different positions. Since long sentences were less frequent, there are less data points for the later positions in each construction. The plots therefore omit positions that cover less than 10% of the data compared to the initial position.

Underneath the bar plots, I give an example that fits the number of plotted words, that is, a sentence that is as long or longer than 90% of the tested sentences of a construction type but shorter than 10% of the sentences of this construction. The noun phrases in these examples are colored according to the correct semantic role for this noun phrase.

5.2.4 Estimating maximal performance

Early in sentences, semantic role assignment was not deterministic since the available information typically was not sufficient to disambiguate the sentence. In light of this temporary ambiguity, I estimated the best possible performance on each position for a given training set. First, I collected all sentence prefixes up to this position from the training set and replaced each word by its category label. I then determined the maximum performance at a sentence position to be the frequency of the most common semantic role for its prefix. Maximum performance for this position was then estimated as the average over all sentence prefixes, weighted by the number of occurrences of each prefix. Tables 5.3 shows an example in a simplified language. The left Table shows a set of sentences with the correct semantic roles for the first noun phrase. It shows the prefixes of length 2 and the corresponding reduced prefix given the noun categories as described in 2.3. The right Table shows how the maximum possible performance given a reduced prefix can be less than 100% in case of ambiguity.

sentence	correct role	prefix(2)	reduced prefix(2)
the girl sleeps	AGENT	the girl	the girl
a cat sleeps	AGENT	a cat	the cat
a dog sleeps	AGENT	a dog	the cat
the cat falls	PATIENT	the cat	the cat
a ball falls	PATIENT	a ball	the ball

reduced prefix(2)	#	AG	PAT	max % prefix
the girl	1	100%	0%	100%
the cat	3	66%	33%	66%
the ball	1	0%	100%	100%
max est.		$\frac{2\cdot100\%+3\cdot66\%}{5}$	$=$	**80%**

Table 5.3.: Simplified example for determining maximum performance estimate. The left Table shows all sentences in a toy language with the correctly assigned semantic role, the sentence prefixes for position 2 and the prefixes with reduced lexical variability in position 2. Here cat and dog are in the same noun category while girl and ball are in different categories. The right Table shows the number of sentences (#) corresponding to each reduced prefix, the percentage of correct roles and the resulting maximum percentage of correct assignments for this prefix (AG=AGENT, PAT=PATIENT). Below, max est. indicates the maximum performance estimate for position 2 weighted over all prefixes.

The maximum performance for position 2 is then determined by a weighted average.

Note that this estimate of maximum performance is optimistic. The actual performance that the NBL model can reach given the input is smaller than this estimate. I assumed correct identification of all word categories in the language and available statistics over a set of sentences with known semantic roles. The NBL model however, was tested on novel sentences with prefixes that it had not necessarily encountered in the training set which means that the statistics over the prefixes were not always known. Furthermore, the estimate is position-specific, that is, it allows independent weighting of cues for each position in the sentence. The NBL model, in contrast, does not change its readout weights depending on the sentence position. Thus, it cannot use cues differentially in different parts of a sentence but has to use a readout that fits all sentence positions. Nevertheless, the estimate is closer to the expected performance than assuming that 100% performance is possible at each sentence position. For example, the estimate can be as low as 38% for the first position of the second noun phrase readout.

5.3 Results

5.3.1 Performance of NBL model and benchmark model

Figure 5.2 shows the performance overall and by word position for each noun phrase. All the performances were kappa corrected (see section 2.4), i.e., the raw model performance was projected into an interval between the performance of a random

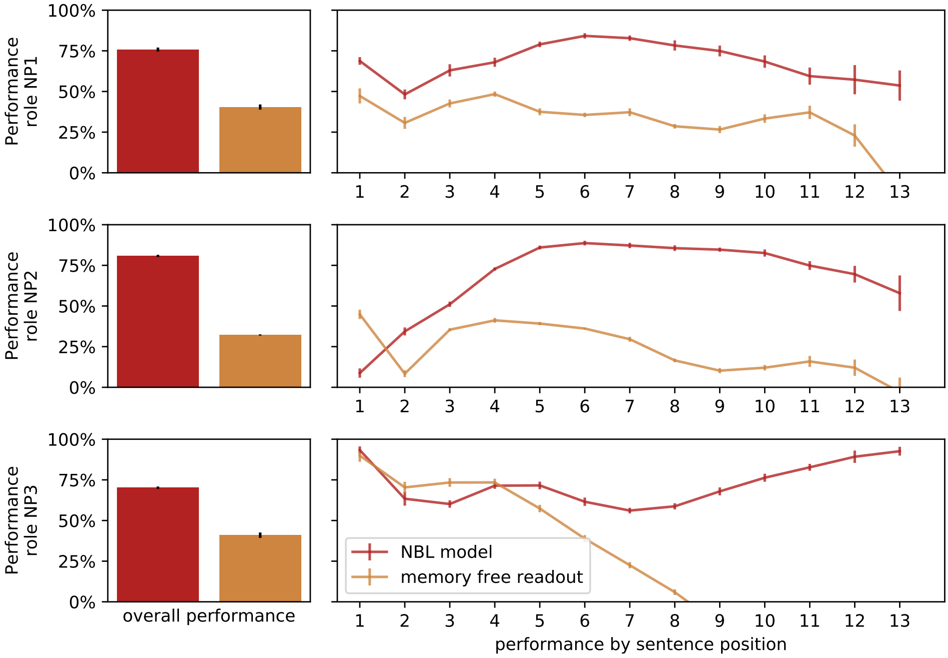


Figure 5.2.: Semantic role assignment, overall and by sentence position. Performance of the NBL model and the memory-free readout overall (left) and for each sentence position (right) on each noun phrase (NP). For all three noun phrases, the overall performance of the NBL model is significantly above that of the memory-free readout. However, early in sentences, the latter can perform better. Late in sentences, the performance of the NBL model declined, indicating a limited memory span. Late sentence positions that occurred in less than 2% of all sentences are not shown. Error bars show 95% confidence intervals.

classifier (interpreted as 0%) and the maximum estimate (set to 100%). To put the NBL model in context, it was again compared to a *memory-free readout model*, where the regression is directly applied to the input sequences. This readout has no memory since it operates on the current input word only. Therefore, the memory-free model shows how well the semantic roles can be inferred from lexical information alone. The difference between the two models shows how much memory the neural network of the NBL model contributes to solving the task.

I tested for differences between models without kappa correction using a paired-samples t-test and found significant differences for each NP (each $t(9)>23$, $p<0.001$). Thus, the overall performance of the NBL model was higher than that of the memory-free readout for all three NPs. I also tested the NBL model compared to the random classifier and found significant differences for all three NPs (each $t(9)>144$, $p<0.001$). The right panels of figure 5.2 show that early in sentences, the NBL model was closer to the memory-free readout than towards the end. This is explained by the fact that the sentence-initial roles are not as dependent on past information as the later ones. Furthermore, the memory-free readout dropped below chance level for late words in the NP3 readout. This may be because the readouts were trained over all words, not by sentence position, and it seems that the memory-free readout optimized towards good performance in early words while sacrificing performance on late words. Overall, however, the memory-free readout was above chance level. In the NBL model, performance was declining towards late positions for NP1 and NP2 even though these semantic roles became increasingly better determined by previously received information. This suggests that network memory could not maintain relevant information sufficiently long and traces began to decay towards the end of sentences. This behavior was not observed for the NP3 readout where the correct assignment often depended on information occurring late in the sentence.

5.3.2 Context-dependent processing

Words are typically not processed in isolation but embedded in sentence or discourse context. Likewise, words only assume semantic roles in the context of other words. This was also the case in the language used here. For example, nouns such as *man* could occur in the same syntactic role in a passive agent-patient transitive, an active theme-experiencer transitive or a double object dative, but assume different semantic roles in each case (see Figure 5.3 B, C, and D). In the first construction, its semantic role was AGENT, in the second it was EXPERIENCER and in the third case, it was the RECIPIENT. This information could not be extracted from the word *man* itself but the linguistic context was required to assign the correct semantic role. The context that was provided in this example was both the verb identity and auxiliaries marking passive voice. Figure 5.3 shows that the NBL model was able to integrate these different linguistic cues and it assigned the correct semantic role in 98±4% (B), 89±3% (C) and 54±2% (D) of the cases.

The full-sentence semantic role assignment required noun phrase specific readouts that were independent from each other. Since most nouns could occur in different noun phrases, the readout for the first noun phrase, for example, could not rely on the mere existence of an input such as *man* but it also had to evaluate if the noun had occurred in the first noun phrase. Figure 5.3A and B shows that this is done correctly in the NBL model on the example of active and passive transitive constructions. For the active transitive, the readout for the first noun phrase identifies the semantic role of *man* while for the passive transitive it identifies the semantic role of *ball*. The

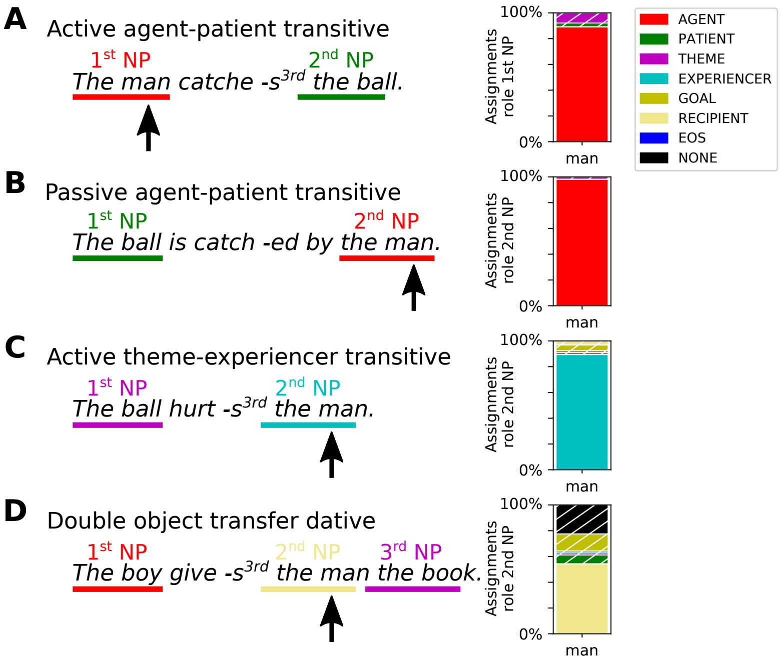


Figure 5.3.: Semantic role assignment is dependent on context. The same noun *man* assumes three different semantic roles in four distinct constructions. Solid bars show that these roles were mostly assigned correctly in each case, during the presentation of the word itself, based on contextual information. Mean frequencies over ten model subjects are shown, error bars were omitted.

reverse is true for the second noun phrase readout. Both assign the correct role to *man* in 89±3% and 98±4% of the cases, respectively.

5.3.3 Incremental integration and prediction

Sentence processing proceeds incrementally as comprehenders do not wait for the end of an utterance but adopt a partial interpretation based on the available information. New words can trigger an update or revision of the interpretation constructed thus far. At the same time, the language system makes predictions about upcoming words and sentence structure. I tested whether the NBL model was able to incrementally revise its interpretation of an unfolding utterance.

In Figure 5.4, the semantic role assignment to three NPs is shown for two of the dative constructions used in the input language. Panel 5.4A shows that evidence for a particular interpretation was accumulating over time. Eventually, the readout settled on RECIPIENT for the second NP and on THEME for the third NP. I will use the

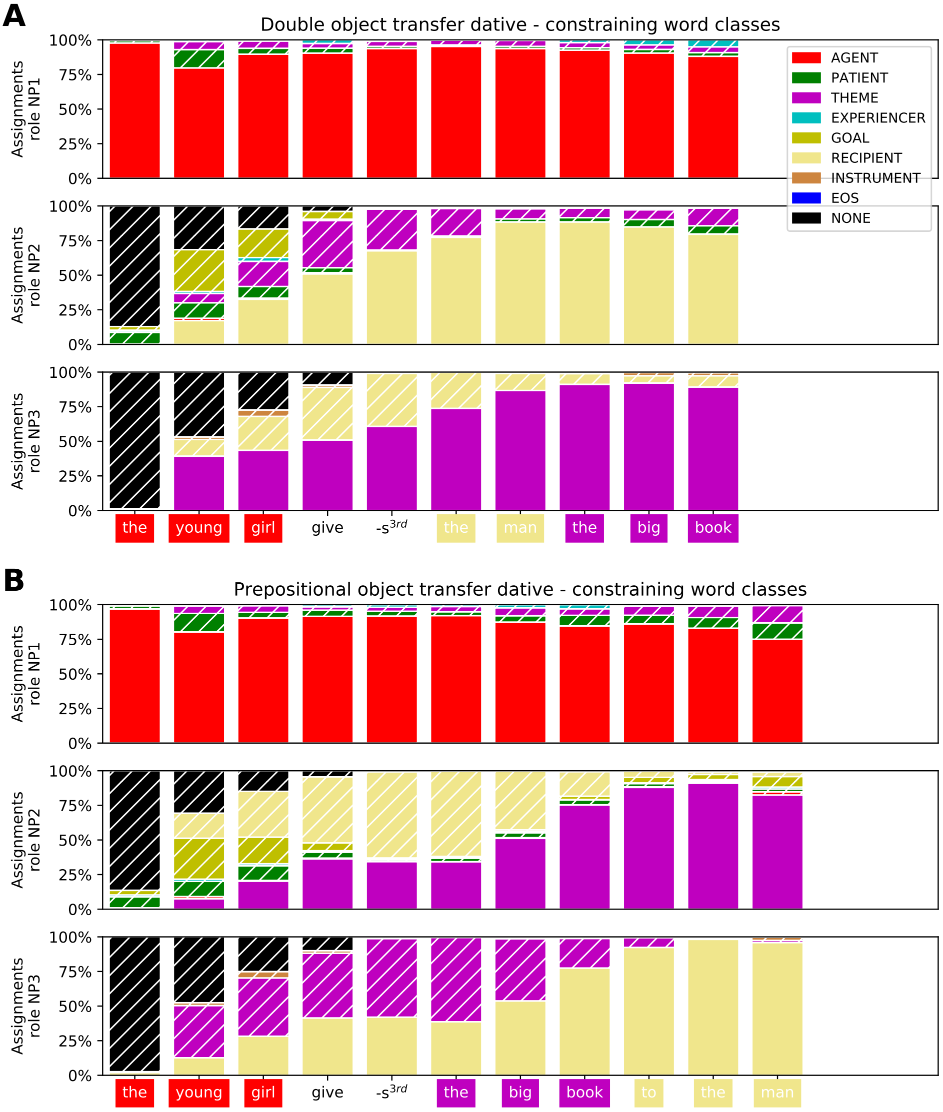


Figure 5.4.: Incremental and predictive role assignment. Semantic role labeling for the two dative constructions that can only be distinguished after the verb. Each panel displays the mean frequencies of semantic roles assigned per sentence position in the two constructions (ten model subjects). Error bars omitted.

shorthand notation NP2:RECIPIENT, NP3:THEME, etc. to describe this assignment. Early in the sentences, several different roles were assigned to NP2 and NP3 before the presentation of the verb *give* (2nd to 6th position). In the input language, *give* belongs to the DATIVE verb category that was only used in the two dative constructions shown in Figure 5.4. The semantic roles for NP2 and NP3 are flipped in the dative alternation; for the double object, NP2:RECIPIENT and NP3:THEME, but NP2:THEME and NP3:RECIPIENT for the prepositional dative. In both constructions, the semantic roles projected by the readout after the verb are reduced to the two options THEME and RECIPIENT. The distinction between the constructions can only be made later based on the animacy of NP2 and the occurrence of the preposition *to*. This happens around the 7th position where the correct role assignment is becoming dominant for each case. This behavior shows both incremental processing and that predictions are made by the NBL model. First, the identity of the verb limits the plausible interpretations and the decision between these two interpretations is made when the relevant information becomes available. Due to the task design, the NBL model also predicts possible sentence continuations at the verb. The proportion of roles for NP2 and NP3 from the 6th position on-wards (the latest point when the verb can occur) that was not either RECIPIENT or THEME was smaller than 15% in each construction. Towards the end of the sentence, when the construction is completely disambiguated, performance was above 80%. The plots also show that relevant information was maintained throughout the whole sentence. In Figure 5.4A, the first word was assigned the AGENT role. This is because it is the most common semantic role of the first NP, reflecting an agent-first bias in English. This assignment is confirmed later by new input and therefore is maintained throughout the sentence which requires processing memory. This memory was provided by neuronal adaptation in the spiking network (see chapter 4).

To further investigate ambiguity resolution, I tested the NBL model on items that required reanalysis late in the sentence. For this, I used sentences that initially triggered the NP1:AGENT, NP2:PATIENT role assignment, which later had to be revised towards NP1:PATIENT, NP2:AGENT with accumulating information. These sentences are similar to garden-path sentences. To test this, I introduced a construction during testing that did not occur in the training set. Sentences from this construction had a semantically unconstraining initial segment (see section 2.3) and were ambiguous between past tense and participle forms of verbs. The examples below show two sentences from the model 's input language, (A) and (C), and the newly introduced form (B) for testing.

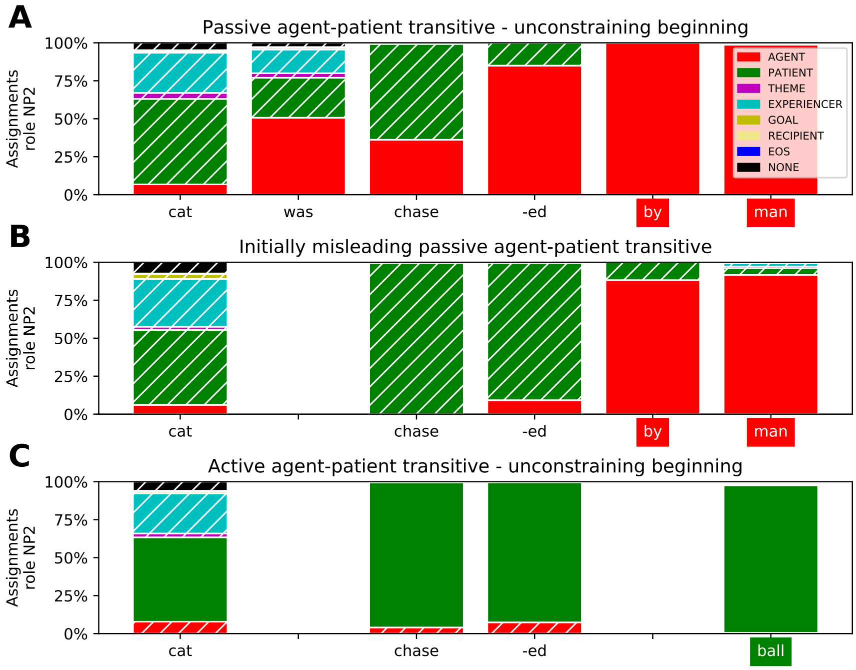


Figure 5.5.: Reanalysis of semantic roles in ambiguous sentences. Role assignment on NP2 of three constructions that correspond to the example sentences (A), (B) and (C) in the main text. The bar plots are normalized on words class, not on sentence position, that is, *cat* marks the readout on the first noun, *was* the readout on the auxiliary verb, *chase* the readout on the verb, etc. Empty slots indicate that this position did not exist in a given construction. In panels B and C, there was no auxiliary verb, in panel C, there was no *by* before NP2. Other words (adjectives, determiners, etc.) were excluded in this plot. In panel B, the word *by* elicits a reanalysis of the sentence. Prior to that, roles are similar to panel C, afterwards, they are similar to panel A.

(A) *The cat was chase -ed by the man.*
(B) *The cat chase -ed by the man was big.*
(C) *The cat chase -ed the ball.*

Sentence (B) has the same prefix as (C), so the two are indistinguishable up to the 4th position. Figure 5.5 shows the semantic roles assigned to NP2 in the different constructions where the panel label corresponds to examples (A), (B) and (C). The roles assigned in panels B and C are similar at the initial noun, verb and suffix *-ed*. For the construction in panel B, the AGENT role changes from 9±1% to 88±1% when *by* is presented. This constitutes a reevaluation of the sentence. The incorrect assignment preceding *by* is changed to the correct role. In panel A the assignment of AGENT to NP2 is triggered by the auxiliary in combination with verb identity and suffix.

5.3.4 Role assignment informed by semantic constraints

Comprehension uses different sources of information to interpret a sentence, including syntactic cues, semantic constraints such as lexical meaning, and world knowledge (Kuperberg & Jaeger, 2016). To test whether the NBL model could also integrate different sources of information, I used semantic constraints which were expressed as distributional properties of the language. The question was whether these constraints would facilitate sentence-level interpretation. I changed the input language such that there was a pair of constructions that only differed in how semantically constraining the first noun was. Semantically constraining means that a noun is clearly identifiable as animate or inanimate. An animate noun, such as *woman*, implies agency while an inanimate noun, such as *ball*, excludes agency. In the construction with a semantically unconstraining beginning, the first noun was instantiated from a word category that contained both animate and inanimate objects and therefore was less semantically constraining. It could both be the agent of an action or an object that an action was performed on. I here use pets to illustrate these unconstraining nouns. For example, both sentences *He gives <u>the cat</u> to her.* and *<u>The cat</u> chases the toy.* were possible while only one instance of each was possible for semantically constraining nouns.

Figure 5.6 shows semantic role assignment on NP1 for both these constructions. The constraining beginning lead to the correct assignment NP1:PATIENT with 84±1% performance at the second position (subject noun). In contrast to that, role assignment on the subject noun with an unconstraining beginning was only correct in 31±1% of the cases. Using paired-samples t-tests to compare both constructions by position, I found a significant difference in performance for the first 5 positions (each $t(9)>3.6$, $p<0.01$) while there was no significant difference for later positions (each $t(9)<0.75$, $p>0.4$). Since the sentences from each construction were variable in length, the positions of the subject noun and the verb were also variable. The difference in performance for the first 5 positions corresponded to the fact that the verb could not occur later than in the 6th position. Thus, verb identity provided the relevant semantic information for the sentences with unconstraining beginning that was otherwise provided by the animacy of the subject noun.

5.3.5 Systematicity

Since natural language syntax is productive, listeners need to be able to process familiar words within novel contexts. Conversely, the semantic roles of novel words can often be inferred from familiar contexts (e.g., *He kicked the <u>wug</u>*). This ability to

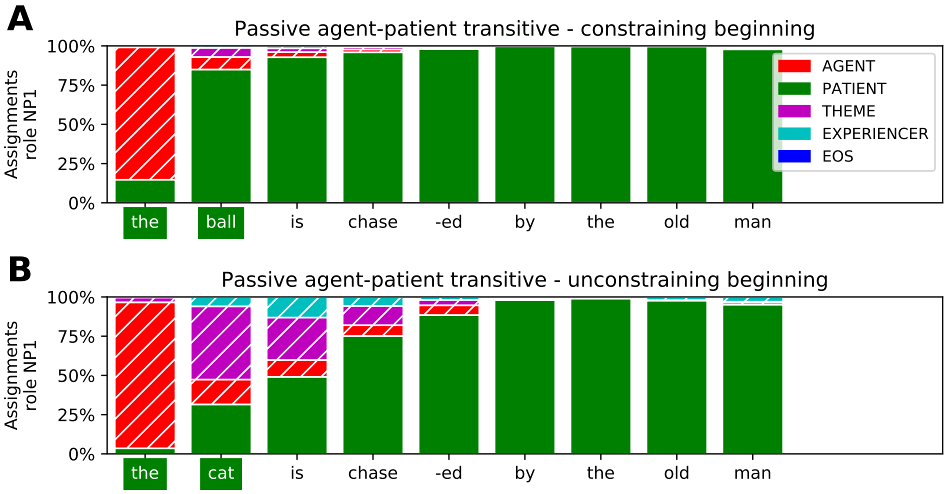


Figure 5.6.: Semantic constraints facilitate early sentence interpretation. Panels A and B show role assignments on two constructions that differed in how semantically constraining the first noun was. In panel A, the first noun is constraining (inanimate object) while in panel B, it is unconstraining. Shown are the mean frequencies of semantic roles assigned per sentence position. Error bars omitted but were never larger than 2%.

generalize beyond experience has been labeled systematicity (Hadley, 1994). In this section investigated the degree of systematicity in the NBL model. It is important to note that the spike patterns that were encoding word input to the network were random and unstructured. Thus, the noun *cat* was as similar to the word *mouse* as it was to *by*, *big*, *jump* or *the*. Consequently, there was no shared feature structure between semantically similar nouns, or between familiar and novel nouns, that could be exploited in the semantic role assignment task. The roles of novel nouns and of familiar nouns in novel contexts had to be inferred from the linguistic context itself. Also, note that adjectives in the input language were not predictive of specific roles, that is, they could not be used to infer the semantic role of a novel noun.

The above-chance performance in section 5.3.1 already shows that the NBL model satisfies weak systematicity because it correctly assigned semantic roles to sentences it was not exposed to during the training phase. The input language to the model could generate around $4 \cdot 10^{11}$ distinct utterances, and hence the $1.8 \cdot 10^{4}$ sentences used for training constituted less than 0.00005% of the total language. Furthermore, the test set only contained novel sentences that were not included in the training set, which is the decisive criterion for weak systematicity (Hadley, 1994). To go beyond this, I explicitly tested stronger forms of systematicity. For four constructions, I excluded sentences with one specific noun from the training. Also,

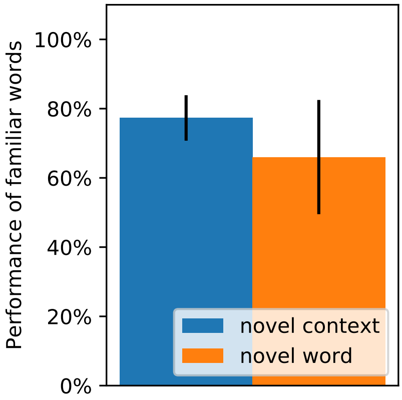


Figure 5.7.: Generalization to novel words and novel contexts. Performance is corrected such that the NBL model performance on a familiar noun in a familiar context is at 100% and a random classifier is at 0%. The novel context condition shows generalization of a familiar word into a novel context. The novel word condition shows generalization to entirely novel words. Error bars show 95% confidence intervals.

other sentences where this noun could occur in the same syntactic position occupying the same semantic role were excluded. For instance, if agent-patient transitives with the word *ball* as object were excluded, such as *The boy hits the ball.*, I also excluded *The boy hits the ball with a stick.* Since the noun *ball* was still used in other sentences, it was familiar to the network but would be tested in a *novel context* (see section 5.2.2 for details). For two other nouns, I excluded all sentences with these nouns from the training input. When tested, these were *novel words* that the network had not been exposed to before. In testing, I included all combinations of nouns and constructions. This allowed me to test how well the NBL model assigned semantic roles in two conditions requiring generalization: (a) familiar words in *novel contexts* and (b) *novel words*. Figure 5.7 shows the performance in both conditions, projected to between the performance on a familiar word and a random classifier. With this correction, I found that performance in novel contexts was at 77±7% and novel words were at 66±16% Using paired-samples t-tests, I tested whether the NBL model performed better than chance. For the novel context condition, I found a significantly higher (uncorrected) performance of 80±5% compared to random baseline with 13±0% (t(9)=26, p<0.001). Similarly, for the novel word condition, I found a significantly higher (uncorrected) performance of 70±14% compared to the same random baseline (t(9)=8, p<0.001). I did not find a significant difference between the two conditions (t(9)=1.7, p=0.12).

These results suggest that the NBL model generalized equally well in both cases, from familiar to novel words and from familiar to novel contexts.

5.4 Discussion

In this study, I found that the NBL model shows a number of characteristics typical of human sentence processing. Words are processed within their sentence context, sentences are processed incrementally, and new information is integrated with existing information. This information is then used to predict the semantic properties of upcoming input. I also found that syntactic cues and semantic constraints were both used concurrently to identify semantic relations between words. When faced with temporary ambiguities, the model assigned semantic roles based on the available cues until new evidence was presented and the interpretation had to be revised. Finally, I found that it exhibited some level of systematicity. It showed weak systematicity since the tested sentences were distinct from the sentences the model was trained on. It also showed a stronger form of systematicity since correct semantic role assignment to familiar words in a novel context, and to entirely novel words, was significantly above chance. These results show, that the level of biological plausibility in the NBL model was sufficient to display a series of cognitively relevant processing features.

Some of these effects have also been obtained in cognitive models of sentence processing before, for example in the Sentence Gestalt model (St. John & McClelland, 1990). This model used an artificial neural network with an input, a hidden and an output layer, where the weights between the nodes were learned through backpropagation in order to assign semantic roles to sentence constituents exploiting both semantic and syntactic constraints. However, the language corpora used in this work contained less than 3000 distinct utterances and the models were trained on 100 000 to 700 000 items. Thus, the ratio of training size to the number of possible utterances was at least 30. In the NBL modelsimulations, I used a language that generated orders of magnitude more distinct utterances and a smaller training set, such that the ratio of trained to possible utterances was less than 10^{-7}. This illustrates the strong generalization capabilities of the NBL model. St. John and McClelland (1990) also showed that the Sentence Gestalt model could generalize to new sentences but they did not test on known words in novel syntactic positions or on novel words (see also Hadley, 1994). Furthermore, their network received event structure as input and target semantic features in training, both of which are not present in spoken or written language, and this was not the case in our simulations. However, the main novelty of the NBL model is that it maps neuronal activity and information processing to a higher degree of biological plausibility and still produces these sentence processing effects. The Sentence Gestalt model used simple rate nodes that were updated in discrete time, while the NBL model implements neuronal and synaptic dynamics in continuous, physical time.

In the introduction of this chapter, I described a set of key features – partially overlapping with Demberg and Keller (2019) – that I tested the NBL model against. Here, I discuss the features from Demberg and Keller (2019) that I did not test and how they relate to the NBL model. They identified four other properties of human sentence processing that cognitive models should reflect. First, (i) processing is fast and robust against grammatical errors and (ii) processing has broad coverage, in other words, it can deal with a variety of syntactic constructions and is not restricted to a particular domain or input modality. Furthermore, they argue that (iii) while a sentence is being processed, words are attached to the same syntactic structure instead of unconnected fragments, and this is referred to as connectedness (Sturt & Lombardo, 2005). Last, (iv) the processor operates with limited memory such that there is a distance-based processing cost. While I did not explicitly test these properties, my findings are not in conflict with them. By design, processing in the NBL model is fast since the readout is done immediately after a word has been presented. The language used here had a variety of different constructions with syntactic alternations, but it remains to be tested how the model deals with a more complex grammar (e.g., embedded clauses). Similarly, it would be an interesting extension of this research to test how the model can work across domains and input modalities. To test connectedness in the model, a formal description and a suitable readout would have to be developed. Finally, I found that the NBL model operates with limited memory determined by the parameters chosen, as shown in chapters 3 and 4, that is, there is a distance-based processing cost. In section 5.3.1 I found that performance declined for later sentence positions. This suggests that early information was not maintained long enough to always yield high performance.

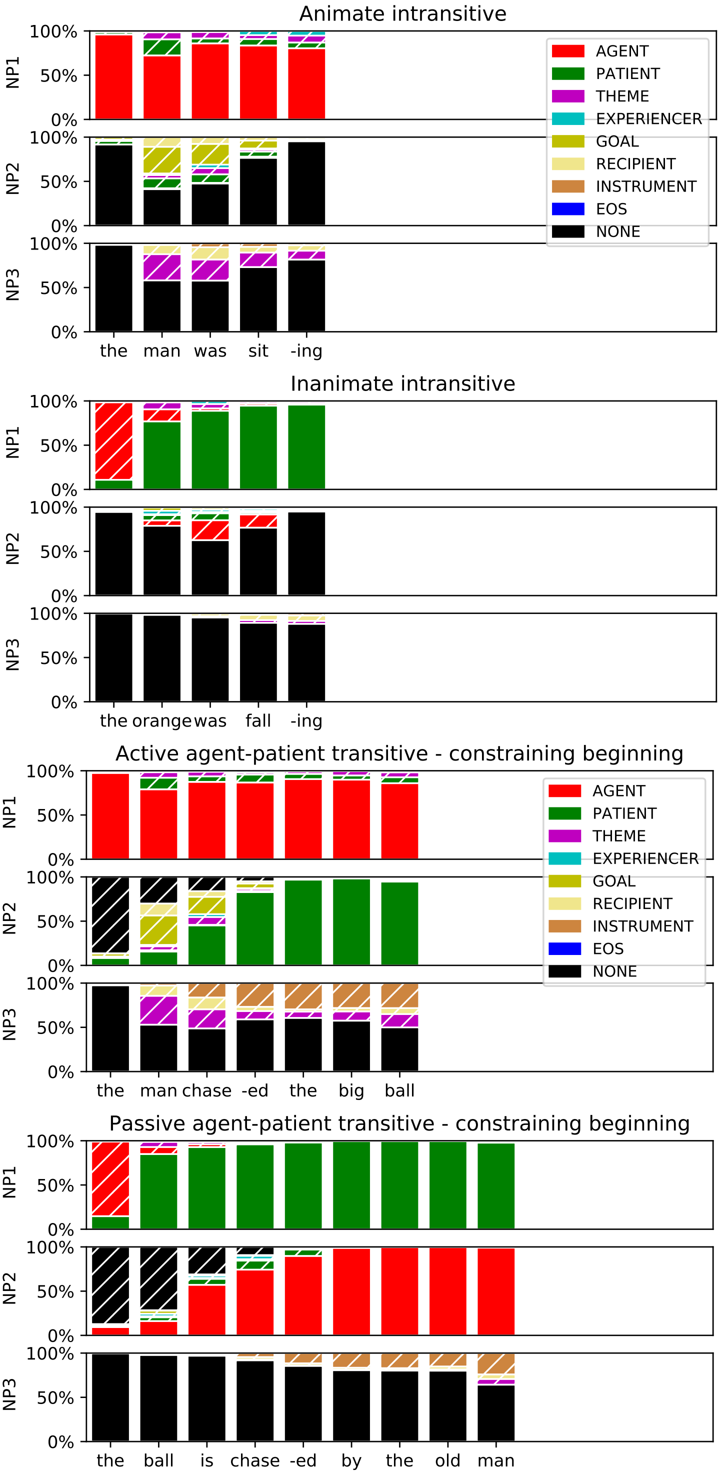
As mentioned in the introduction, I used "prediction" in the minimal sense that the state of the processor is influenced by context such that information about an upcoming item is implicitly available before the actual input (Kuperberg & Jaeger, 2016). The predictions made by the NBL model are graded rather than all-or-nothing predictions. For example, in Figure 5.4, similar sentence prefixes from two equally likely constructions elicited two possible predictive interpretations roughly half of the time. This was the case even though, for each word in a sentence, I used a winner-takes-all selection criterion and only chose the most active prediction. Across all sentences of a construction, however, there was a gradual change of prediction which indicates that for local ambiguity each sentence prefix triggered multiple interpretations. The correct role assignment was then determined by the concrete words in a given example. Importantly, in the NBL model, "predictions" about past or future items are made using the same mechanism. The readout assigns semantic roles to noun phrases independently of whether they are still upcoming or whether new information causes a reevaluation of previous noun phrases. It is conceivable that the human language system uses a similar mechanism to generate implicit predictions about upcoming input but also uses this mechanism to reevaluate past input in light of new, disambiguating information. The NBL model is able to integrate different sources of information in sequential processing to inform semantic role assignment, for example, word identity, morphosyntax, serial order, distributional semantics, et cetera, and this did not require specialized mechanisms to cope with different types of cues. All this information was continuously folded into the network in a state-dependent manner and from this state semantic relations could be inferred reliably. This is in agreement with experimental findings that human sentence processing integrates linguistic information from different levels to inform different levels of processing. Hagoort et al. (2004) found that it is not necessarily possible to distinguish the different processes responsible for the integration using neuroimaging methods.

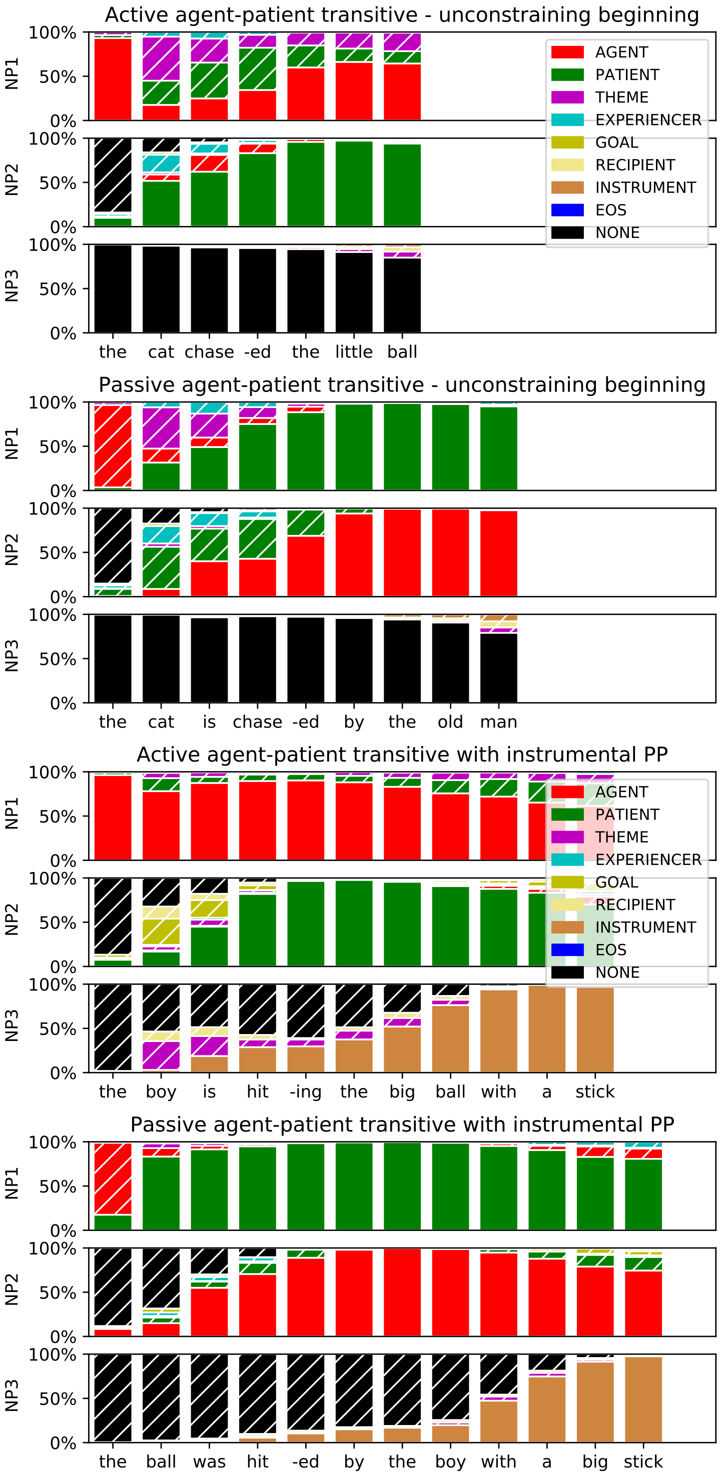
The forms of systematicity investigated in this chapter are not a full exploration but still serve as an indicator of the potential of the NBL model. For instance, I did not test systematicity in terms of generalization to embedded clauses, as was required by Hadley (1994). However, according to Hadley, even for single-clause utterances most connectionist models fail to exhibit strong systematicity unless they make strong assumptions. In particular, he criticized models that rely on shared features between familiar and novel words to facilitate generalization. Furthermore, the models discussed by Hadley used a large training set compared to the number of distinct utterances their language could produce. Thus, systematicity in connectionist models remains a contentious issue (Frank, 2006). In my study, words were represented as random spike patterns, frozen noise generated by a series of Poisson processes, and these words did not share features in a systematic fashion. In training I only used a small set of sentences compared to the number of possible utterances that the input language could generate. Hence, for single-clause utterances, the NBL model exhibited a strong level of systematicity. The architecture for which strong systematicity has been shown, and which is most similar to the NBL model , is the echo state network (Frank, 2006). Similar to my model, they used a static recurrent network in conjunction with a trained readout, however, they did not use spiking neurons or spike-pattern input.

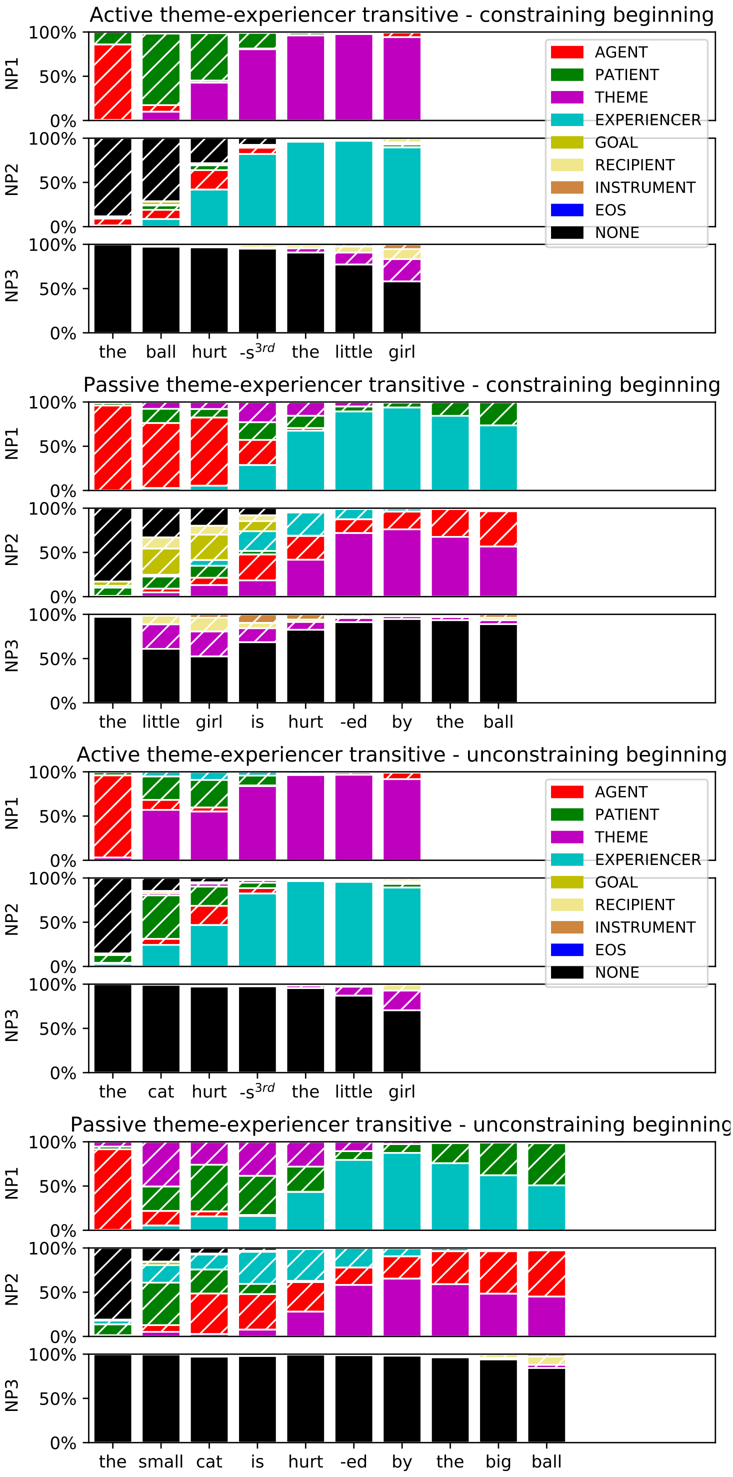
In conclusion, I found that our current understanding of biological neurons and the ability to simulate spiking networks computationally is sufficient to connect to cognitively relevant aspects of language processing. Thus, a point has been reached where computational neuroscience, which has been investigating neural processing and circuit dynamics for decades, can be integrated with the cognitive sciences in a manner that is beneficial to both fields and mutually facilitates research progress.

5A Appendix

In the results section of this chapter, I only discussed a selection of sentence constructions from the input language. For transparency, I append the complete role assignment plots for all of the constructions.


Animate intransitive
NP1
NP2
NP3
100%
50%
0%
the
man
was
sit
-ing
AGENT
PATIENT
THEME
EXPERIENCER
GOAL
RECIPIENT
INSTRUMENT
EOS
NONE
Inanimate intransitive
the
orange
was
fall
-ing
Active agent-patient transitive - constraining beginning
the
man
chase
-ed
the
big
ball
Passive agent-patient transitive - constraining beginning
the
ball
is
chase
-ed
by
the
old
man

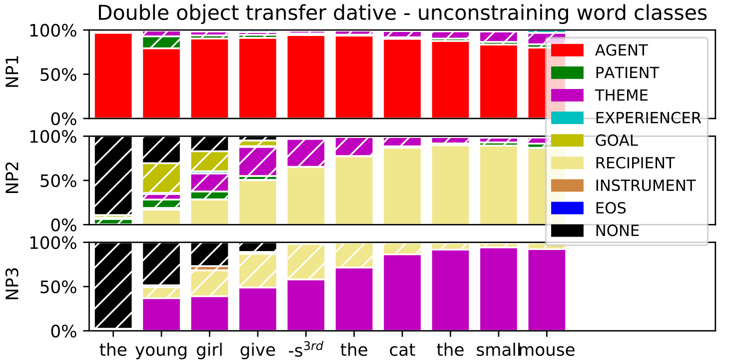

Active agent-patient transitive - unconstraining beginning
NP1
NP2
NP3
100%
50%
0%
the
cat
chase
-ed
the
little
ball
AGENT
PATIENT
THEME
EXPERIENCER
GOAL
RECIPIENT
INSTRUMENT
EOS
NONE
Passive agent-patient transitive - unconstraining beginning
the
cat
is
chase
-ed
by
the
old
man
Active agent-patient transitive with instrumental PP
the
boy
is
hit
-ing
the
big
ball
with
a
stick
Passive agent-patient transitive with instrumental PP
the
ball
was
hit
-ed
by
the
boy
with
a
big
stick

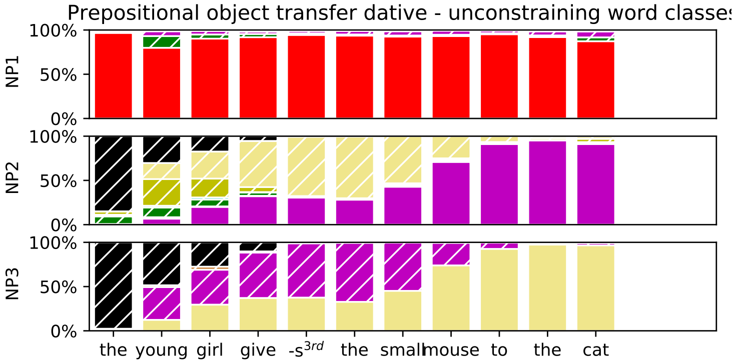

Active theme-experiencer transitive - constraining beginning

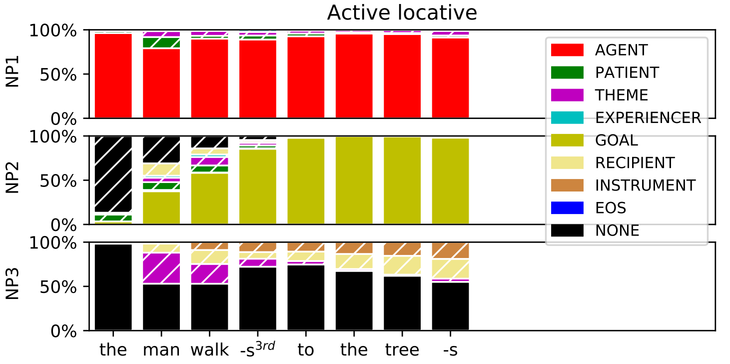
Passive theme-experiencer transitive - constraining beginning

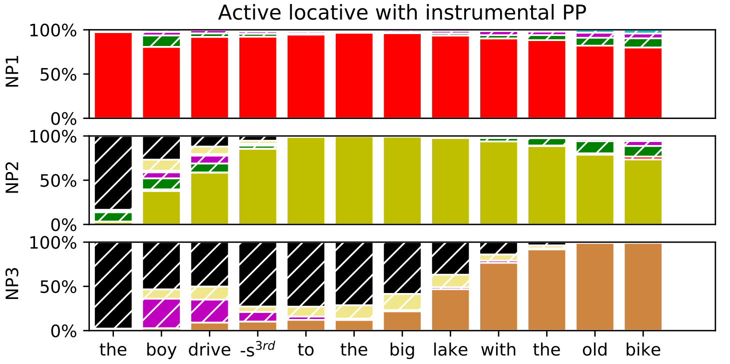
Active theme-experiencer transitive - unconstraining beginning

Passive theme-experiencer transitive - unconstraining beginning


Double object transfer dative - unconstraining word classes
NP1
NP2
NP3
100%
50%
0%
AGENT
PATIENT
THEME
EXPERIENCER
GOAL
RECIPIENT
INSTRUMENT
EOS
NONE
the
young
girl
give
-s3rd
the
cat
the
small
mouse


Prepositional object transfer dative - unconstraining word classe
NP1
NP2
NP3
100%
50%
0%
the
young
girl
give
-s3rd
the
small
mouse
to
the
cat


Active locative
NP1
NP2
NP3
100%
50%
0%
AGENT
PATIENT
THEME
EXPERIENCER
GOAL
RECIPIENT
INSTRUMENT
EOS
NONE
the
man
walk
-s3rd
to
the
tree
-s


Active locative with instrumental PP
NP1
NP2
NP3
100%
50%
0%
the
boy
drive
-s3rd
to
the
big
lake
with
the
old
bike

6

Binding in the NBL model

6.1 Introduction

The binding problem in its general form is concerned with the question how distinct pieces of information are combined and integrated by the brain such that a limited number of conceptual primitives stored in our memory can be used to represent and understand the possible states of our environment. Feldman (2013) distinguishes four problems from different fields of cognitive science that all are referred to as "binding problem" but are distinct and should be studied separately; general coordination, unity of perception, visual feature-binding, and variable binding. Of these versions, variable binding is the most relevant for language processing and this will be my focus. Importantly, the term "problem" does not describe a problem for the brain but the fact that it is scientifically not well-understood how binding is accomplished (Feldman, 2013).

Binding in language processing is a pervasive phenomenon that occurs at different linguistic levels. Jackendoff describes this as the "massiveness of the binding problem" (Jackendoff, 2002). For example, when processing a spoken utterance, in order to integrate the phonological, syntactic and semantic structure, it is necessary to temporarily store the syntactic relationships between each sentence constituent and to connect the speech sounds to conceptual representations. In a sentence like *The black cat is chasing the red toy*, binding between nouns and semantic roles is required to comprehend *cat* as AGENT and *toy* as PATIENT. Furthermore, binding between nouns and modifying adjectives marks the cat as black and the toy as red instead of the other way around. These are just a few of the binding relations necessary to interpret a sentence correctly. In a sentence of similar complexity, Jackendoff (2002) identified upwards of a hundred binding operations required for a full analysis.

The binding problem has been intensely debated in relation to cognitive models of language processing. That is because with traditional connectionist models it is difficult to implement and explain binding as described above and it has been suggested to be a fundamental shortcoming compared to symbolic models of language processing (Fodor & Pylyshyn, 1988). Symbolic models use variables that can be freely manipulated and combined using mathematical and logical operations. In contrast to that, connectionist models represent information as distributions of network activity which are processed through the interaction of the nodes. The difficulty to represent binding relations led some to conclude that they are ill-suited for modeling language processing (Fodor & Pylyshyn, 1988; Jackendoff, 2002). However, it is a fact that the brain, which in a broad sense functions like a connectionist network, is able to solve the binding problem. In the words of Pulvermüller (2010):

> “This critique seems, however, to be aimed at neural networks of a certain kind. Applied to the brain itself, it would lead to the apparent paradox that the language organ of humans cannot process human language – an irrational position.”

In response, Hummel (2011) identified three coping mechanisms that he observed: (i) argue that mental representations are symbolic, ignoring or down-playing the undisputed non-symbolic character of the neural infrastructure, (ii) argue that mental representations are non-symbolic, omitting the evidence for relational representations necessary for human reasoning, or (iii) attempt to bridge the gap between connectionist and symbolic models by specifying what is necessary to enable connectionism to solve the binding problem (see, e.g., Fitz & Chang, 2009). I take the latter stance and investigate what kind of binding operations can be performed with the NBL model. However, first I review the mechanisms for binding that have been proposed by connectionist models.

6.1.1 Connectionist approaches to solve the binding problem

Broadly speaking, there are four approaches to address the binding problem in connectionism (Sougné, 2006; Hummel, 2011). Two of them have obvious issues and are mentioned here only to further illustrate the problem while the other two are promising proposals that continue to be investigated. To describe these approaches, I distinguish between *primitive* and *composed* concepts. A primitive is whatever is stored in long-term memory and the smallest unit of information in a connectionist model, for example a single activated node. A composed concept is not stored in long-

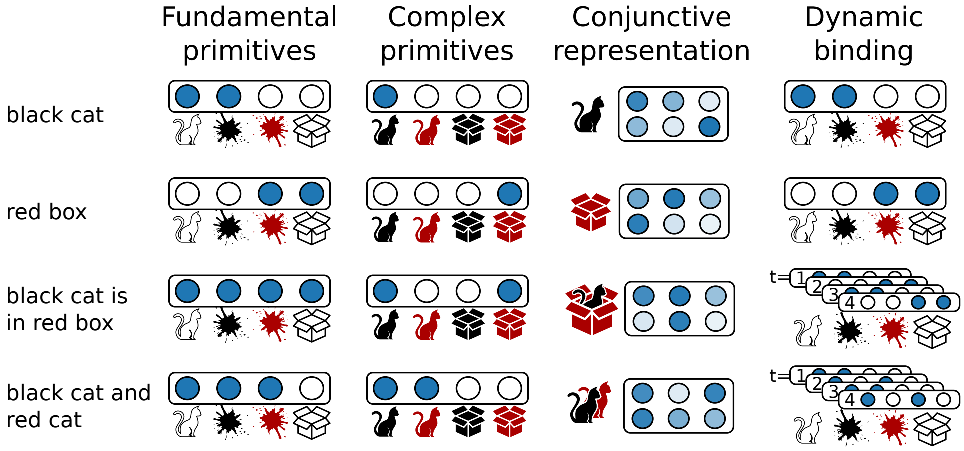


Figure 6.1.: Connectionist approaches to the binding problem. The columns illustrate the four approaches described in the main text. For each, the representation of composed concepts corresponding to the utterance on the left is shown. The third row displays representations of utterances that contain two distinct composed concepts. When primitives repeat, the problem of 2 occurs, and this is shown in the fourth row. Note that the representations for the bottom two items in the fundamental primitives column do not allow unambiguous reconstruction of the input.

term memory; its representation requires a combination of primitives, for example by activating several primitive nodes simultaneously. Examples of the four different approaches to binding are shown in Figure 6.1.

(1) Use *fundamental primitives*, that is, separate and independent nodes for each conceptual feature, such as “black”, “red”, and “cat”. This allows using a small set of primitives that can be flexibly combined into composed concepts such as “black cat” and “red box” by co-activating several primitives. However, such an additive representation can cause confusion in cases like “the black cat is in the red box” which would be indistinguishable from “the red cat is in the black box”, as shown in Figure 6.1, third row.

(2) Use *complex primitives*, that is, separate and independent representations for each possible combination of conceptual features. Thus, there are complex primitives for “black cat” and “red box” as well as for “red cat” and “black box”. This solves the problem of how adjectives are attributed correctly to the respective nouns. However, this leads to a combinatorial explosion in the number of primitives. Each of them represents a composed concept and needs to be stored separately. Moreover, there are no shared properties between “black cat” and “red cat” making generalizations from one to the other difficult.

Another variation on complex primitives is to use fundamental primitives but have a dedicated node for each possible binding between them, such as in van der Velde and de Kamps (2006). These models suffer from an explosion as well, but now in the number of nodes that represent the possible bindings (Hadley, 2006; Dyer, 2006).

I now turn to more promising approaches towards solving the binding problem:

(3) Use of *conjunctive representations* where primitives are stored as multidimensional concept vectors that can be combined to generate new but related representations of composed concepts. The first approach of this kind used the tensor product to combine two or more vectors into a new, higher-dimensional representation of a composed concept (Smolensky, 1990). The tensor product is order-sensitive, such that "the black cat in the red box" will have a different representation than "the red cat in the black box". This addresses some of the problems with the approaches (1) and (2) above. There are variations on this method in order to avoid the increase in dimensionality incurred by using tensors, for example, through holographic projection (Plate, 1995) or circular convolution in vector symbolic architectures (Gayler, 2004; Eliasmith et al., 2012).

(4) Use of *dynamic binding* that exploits the temporal nature of sequences of items and represents binding relations dynamically. One such implementation achieves binding through synchrony (Shastri & Ajjanagadde, 1993) which can use fundamental primitives as in (2), but at any point in time only co-activates the nodes that are to be bound. That is, when encountering "the black cat is in the red box", at time 1 "black" and "cat" are active, while at time 2 "red" and "box" are active together, and so forth (Figure 6.1). To obtain a stable representation, the synchronous activations are replayed periodically, creating oscillations where all binding relations are cycled through one-by-one. This mechanism, again, addresses some of the problems in (1) and (2).

The dynamic binding approach (4) has gained traction since there is some supporting evidence from visual processing (e.g., Eckhorn et al., 1988), and several proposals have used "synchrony" to realize binding in connectionist models (Sougné, 1998, 2001; A. E. Martin & Doumas, 2017; Rabagliati et al., 2017). Sougné (2001) is particularly relevant here, since they implemented synchronous binding in spiking neural networks. They used neurons that fire with a fixed period when activated, and Hebbian learning mechanisms were engaged to synchronize and desynchronize the neurons to realize the binding of roles and fillers. This proposal assumes that there are neurons which function as intrinsic oscillators that are dedicated to binding. There is, however, little evidence that such neurons exist, and thus it is unclear how periodic reactivation is implemented and coordinated. Apart from this, it is unclear how synchronous activation is used for processing by downstream regions (van der Velde & de Kamps, 2015). Furthermore, Jackendoff (2002) has questioned the viability of this approach in light of the massiveness of the binding problem. To process language, many binding relations have to be established simultaneously, which raises the issue of how long the oscillation periods are, and how many parallel binding relations can be represented and maintained within that time window.

An alternative implementation of dynamic binding that is directly based on the properties of spiking neural networks is *polysynchrony* (Izhikevich, 2006). There, groups of spiking neurons form a spatio-temporal ensemble such that once the initial group members are activated, a temporally extended spike pattern is triggered that can encode binding information. It can be formed through spike-time dependent plasticity and relies on variable spike transmission delay times. This approach has been successfully used to model aspects of visual binding (Isbister et al., 2018).

6.1.2 Problem of 2

One aspect of the binding problem that is particularly difficult to handle for connectionist models is the problem of 2, also referred to as the multiple instantiation problem, or type-token problem (Sougné, 2006). Jackendoff (2002) identified it independently from the binding problem as one of the four big challenges for connectionism. It occurs in cases when the retrieval of a primitive from long-term memory happens more than once during the processing of a sentence. For example, in *The black cat is chasing the red cat*, binding is not just necessary to process the adjectives and nouns correctly but also to conceptually distinguish both instances of *cat* (Figure 6.1, bottom row). There are different approaches to deal with the problem (Sougné, 2006):

(1) Load several instances of the repeated primitive (here *cat*) into working memory. This is inspired by symbolic models that can have an arbitrary number of items of the same type in working memory, but it is not clear how this could be implemented neurobiologically.

(2) Hold several instances of the repeated primitive in long-term memory. This uses the same rationale as when using complex primitives to address binding and,

correspondingly, has similar issues. If there are n instances of *cat* in long-term memory, then there will be a hard limit of $n+1$ instances of *cat* in a sentence that cannot be processed correctly anymore. It is also unclear if one has the same number of instances for each primitive, how this number is determined, and if it can be changed.

(3) Use a doubling of the frequency to encode the repeated retrieval of the same concept in dynamic binding through synchrony. This relies on periodic reactivation of the primitives. Such a mechanism can repeatedly activate the same primitives at two different points of the oscillation. However, concerns regarding the binding through synchrony remain because it is unclear what causes the reactivation and how it is used for further processing.

6.1.3 Query task and parallel readout

In the NBL model, I operationalized binding in terms of question answering, where an item A is bound to variable B if A can be retrieved by asking "What is the information bound to B?" (van der Velde & de Kamps, 2006). I do this using two different tasks, one that implemented this question answering explicitly, and one implicit task where I show that the relevant information for binding is available. In the explicit binding task, I *query* the network with a semantic role after the presentation of each sentence, such as "AGENT?", which represents the question "Who was the agent in the last sentence?". I then train a readout to map the resulting network state onto the noun corresponding to the queried semantic role. Importantly, I use the same readout for all different queries such as "AGENT?", "PATIENT?" and "THEME?" and this readout has to generate the correct answers.

In the implicit binding task, I decode different pieces of information after the sentence has been presented. I use *parallel readouts*, similar to chapter 5, to determine which semantic roles were assigned to the first/second/third noun phrase. Additionally, I read out the identities of the nouns, adjectives, and verb in the sentence, as well as the tense, aspect, and voice of the event semantics. The parallel readouts show that the relevant information is available in the neural network and can be accessed by downstream networks for further processing. They also show that the binding information represented by the NBL model is not limited to only one instance per sentence, but that it can provide information for several independent binding operations, as required for language processing (Jackendoff, 2002).

6.1.4 This study

In this chapter, I investigate the binding problem using the NBL model. First, I test if the model can bind words to semantic roles using the query task. Second, I investigate if the network can represent a larger number of binding relations simultaneously, that is, whether it is suitable to address the massiveness of the binding problem. I test this using thirteen independent parallel readouts. Third, I use the information provided by the parallel readouts to test if the problem of 2 can be solved by the NBL model, where different instances of the same concept within a sentence have to be identified. I combine the information from the adjective and noun identity readouts to do that. For example, for the sentence *The red cat chases the black cat*, I test if this information connects *red* to the first and *black* to the second instance of *cat*. In the final section, I investigate the state-space dynamics of the network and compare model behavior to the proposed solutions to the binding problem discussed in sections 6.1.1 and 6.1.2 above.

6.2 Methods

I simulated the NBL model as described in chapter 2, informed by the insights from chapters 3 and 4. Table 6.1 lists the parameters I used here which diverge from the general description.

Variable	Value	Description
N_{tot}	2000	total number of neurons
N_{exc}	1600	number of excitatory neurons
N_{inh}	400	number of inhibitory neurons
τ_w	200 ms	neuronal adaptation time constant
τ_{syn}	500 ms	synaptic current time constant
ρ_c	5%	network connection density
f_{tune_n}	5 Hz	tuned target spike rate
n_{train}	200 000 words $\approx$ 13 000 sentences	training set size (including queries)
n_{test}	50 000 words $\approx$ 3100 sentences	testing set size (including queries)

Table 6.1.: NBL model parameter values used in this chapter.

Generally, I found that the query task was more sensitive to network and lexicon size than previous tasks. Thus, I doubled the total number of neurons while keeping the ratio of inhibition to excitation and all other parameters the same (encoding projection, connection density, tuning frequency, etc.). The training set contained 200 000 words which corresponded to approximately 13 000 sentences that were presented during the input phase. I also reduced the lexicon size, thus decreasing the number of possible utterances in the language (see below). The number of training sentences was still less than 0.000 25 % of all possible sentences generated by the language. Furthermore, I found that tuning the network to a target firing rate was not always successful. For instance, when the network was tuned to 5 Hz at the beginning of each simulation, it sometimes deviated from this level of activity at the end. Model subjects were excluded from analysis if the average firing rate was less than half or more than twice the target rate. This happened 3 out of 15 times. Hence, all results in this chapter are averages from 12 model subjects with the same network parameters but different random seeds. These randomized subjects differed in the composition of their training and test sets, the specific synaptic connectivity, and the projection patterns used to encode input to the network.

6.2.1 Querying task

As described in section 6.1.3, I used the query task to test binding in the NBL model explicitly. To reduce the computational resources needed for training and testing, I attached a query for each semantic role in a sentence. For example, a test sentence with the roles THEME and EXPERIENCER would look as follows:

> *The boy was hurt by the ball . THEME? EXPERIENCER?*

The separate input items are separated by spaces. The correct readout response to the query would then produce *ball, boy*, in this order. To avoid that the different queries interacted and influenced each other, I saved the full network state (membrane potentials, adaptive neuronal currents and synaptic currents) at the end of each sentence and reloaded it before each query. In this way, I could use the sentence-final network state and apply different queries to this sentence. All of the queries were presented for a duration of 150 ms. Depending on the number of semantic roles in a sentence, there were one, two or three queries. There were approximately 28 000 queries in the training set and 7000 queries in the test set for each model subject. To calibrate the readout, I separated the network states resulting from the query input from the rest of the sentences, and only trained (and tested) on the states generated by the queries.

Example: *The little girl was give -ing*
the man the heavy book.

Readout	**# options**	**Output**
semantic role 1^{st} NP	4	AGENT
adjective 1^{st} NP	9	little
noun identity 1^{st} NP	44	girl
semantic role 2^{nd} NP	7	RECIPIENT
adjective 2^{nd} NP	9	NONE
noun identity 2^{nd} NP	45	man
semantic role 3^{rd} NP	4	THEME
adjective 3^{rd} NP	9	heavy
noun identity 3^{rd} NP	45	book
verb identity	48	give
aspect	2	progressive
tense	2	past
voice	2	active

Table 6.2.: Target outputs for an example sentence with the readout name (left), number of possible outputs per readout (middle) and the correct output for the example sentence (right).

6.2.2 Parallel readouts

I used parallel readouts to extract different types of information after the presentation of each sentence. The readouts were trained on all words in a sentence. I only report the performance on presentation of the end-of-sentence marker where sentences were disambiguated and the maximum performance was 100%. In addition to decoding semantic roles, I used readouts for other sentence properties such as the identity of words (nouns, adjectives, and the verb). Furthermore, there were three readouts identifying features of the event semantics of a sentence; voice, tense and aspect. These different readouts and their target values are shown in Table 6.2 for an example sentence. In total, there were thirteen parallel readouts trained on the same recorded network states. As in chapter 5, when a readout was not applicable to a sentence, the target output was set to *NONE*, for example when there were less than three noun phrases in a sentence, or when there was a noun without an adjective. To investigate the problem of 2 (Po2, for short), I used the output from these parallel readouts for sentences with repeated nouns in the first and second noun phrase (Po2 sentences) generated by the four constructions described in Table 6.3 (active/passive agent-patient and theme-experiencer transitives).

Construction (frequency)	**Word categories and examples**
Transitive (Agent-Patient)	Action = TRANSITIVE (AGENT-PATIENT); Agent = LIVING; Patient = OBJECT; (Instrument = INSTRUMENT)
Main form (9.5%)	*The man catch-s*^{3rd} *the small apple.*
Po2 form (9.5%)	*The small dog catch-s*^{3rd} *the big dog.*
Syntactic alternation (4.5%)	*The small apple is catch-ed*^{par} *by her.*
Po2 Syntactic alternation (4.5%)	*The small dog is catch-ed*^{par} *by the old cat.*
Instrumental preposition (5%)	*The man catch-s*^{3rd} *the small apple with a net.*
Alt. instrumental prep. (2.5%)	*The apple is catch-ed*^{par} *by the man with a net.*
Transitive (Theme-Experiencer)	Action = TRANSITIVE (THEME-EXPERIENCER); Theme = OBJECT; Experiencer = LIVING
Main form (4.5%)	*The stick scare-ed the man.*
Po2 form (4.5%)	*The cat scare-ed the dog.*
Syntactic alternation (2.5%)	*The man was scare-ed*^{par} *by a stick.*
Po2 Syntactic alternation (2.5%)	*The small cat was scare-ed*^{par} *by a red cat.*
Roles: AGENT, PATIENT, THEME, EXPERIENCER, INSTRUMENT, ACTION	

Table 6.3.: Constructions that were changed relative to Table 2.2. Po2 marks constructions that could generate sentences with repeated instances of the same noun. In these cases, for example, *cat* could occur as an AGENT as well as a PATIENT, THEME or EXPERIENCER.

6.2.3 State-space trajectories

To visualize the state-space trajectories of the spiking network, I recorded the membrane potentials and the adaptive currents for each neuron at every millisecond. The aim was not to perform an analysis of the state-space but to illustrate how the network state moves through this high-dimensional space for a particular sentence or sequence of words. To reduce dimensionality, I applied principal component analysis (PCA) on data from a selected set of keywords that I wanted to emphasize in the respective plot. The PCA shows both the state changes during the processing of each keyword and the changes in the transition between words. These keywords are indicated in the caption of each figure. Then, I used the coordinate system spanned by the first three principal components to plot the trajectory of a sentence or set of words.

6.2.4 Input language

I adapted the language from section 2.3 to enable me to investigate the questions relevant in this chapter. The lexicon size was reduced from 16 to eight items in each category (116 words in total). I also allowed the repetition of nouns and adjectives within a sentence. To identify the different nouns by their adjectives, I increased the probability of an adjective in each noun phrase to 75%. Finally, I changed four constructions such that they generated sentences with multiple instances of the same noun – Po2 sentences. I used the transitive constructions with unconstraining sentence beginnings and changed the second noun to also be semantically unconstraining. A semantically unconstraining noun could occupy both an active role, such as AGENT, and a passive role, such as PATIENT. These were the nouns in the LIVING/OBJECT category (see section 2.3). In these Po2 constructions, a sentence that contained the same noun twice was generated in one of eight cases. Table 6.3 shows the updated language fragment of transitives. All other constructions remained the same. With these changes, the number of distinct utterances that could be created by the language was 5.5 billion.

6.3 Results

6.3.1 Querying binding information

In the *query task*, first a sentence was presented to the model, then a semantic role query was injected into the network, and then the readout had to respond with the lexical noun or pronoun that occupied this role in the sentence. For example, in the test item *The woman gives the book to the boy. AGENT?* the correct readout response to the “AGENT?” query would be *woman*. The results from 12 model subjects, tested on 7000 queries, are shown in Figure 6.2. Panel A shows the overall performance of the NBL model compared to a random classifier. Note that all 44 nouns and pronouns in the language were possible responses. Therefore, the random classifier only performed at 3±0% while the model reached 78±0% binding performance. A paired-samples t-test showed that this difference was significant (t(11)=638, p<0.001). Panel B of Figure 6.2 shows the overall binding performance split by the different semantic role queries. Performance of the NBL model was at least 73% for each queried role, with the “INSTRUMENT?” and “GOAL?” roles both yielding higher performance above 93%. This is likely because the nouns that were filling these roles were always positioned at the end of sentence and there was a dedicated noun category for each of the two roles. In contrast, the random classifier never reached beyond 4% performance. Paired-samples t-tests showed a significant difference between the two models for all distinct role queries (for each t(11)>78, p<0.001). Panel C of Figure 6.2 shows how the incorrect binding responses were distributed for each noun. Since the lexical nouns in the language had distributional semantics due to noun category membership, I labeled the blocks of eight nouns with the

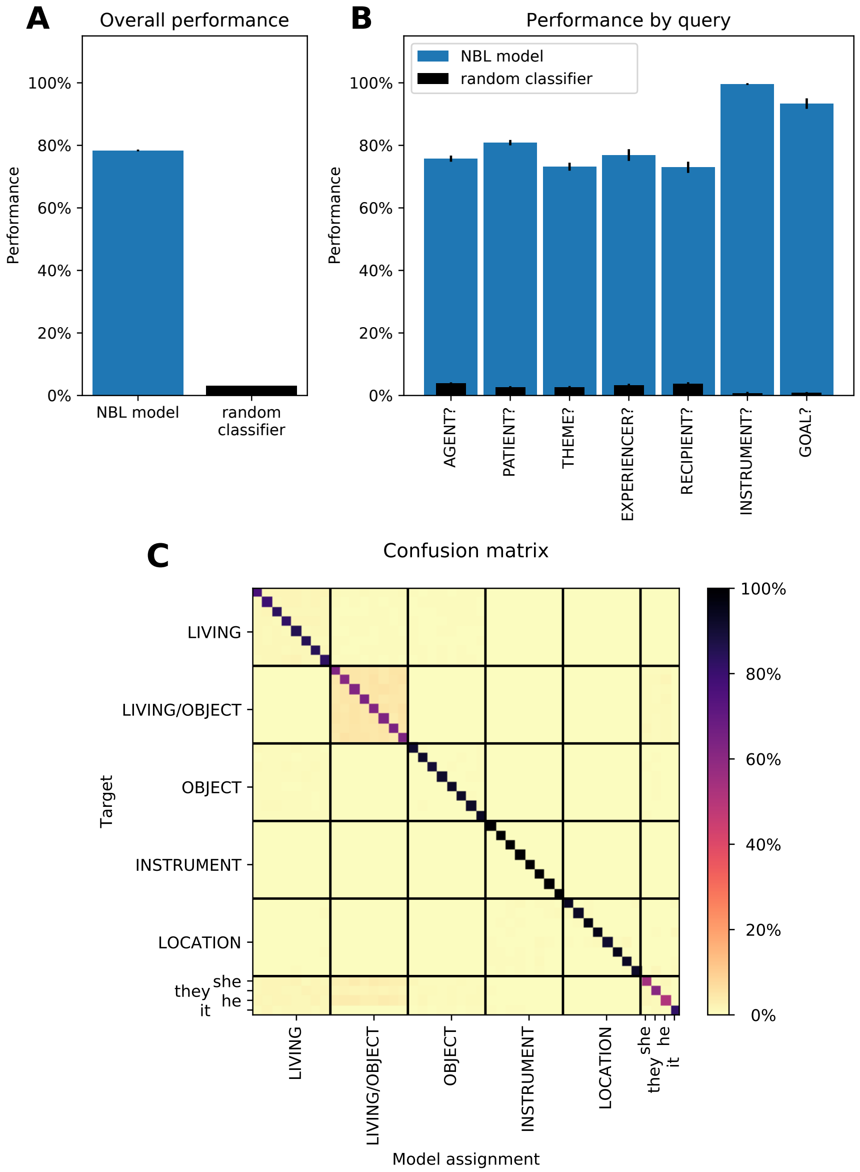


Figure 6.2.: Binding performance in the query task. A: Overall performance of the NBL model compared to a random classifier. **B:** Performance split by the different semantic role queries. Error bars show 95% confidence intervals. **C:** Confusion matrix by noun, grouped into word categories. For each readout binding target, it shows how often the correct word was selected (on the diagonal) and how often other words were assigned (off the diagonal). Most off-diagonal fields had a confusion value close to 0% (light yellow).

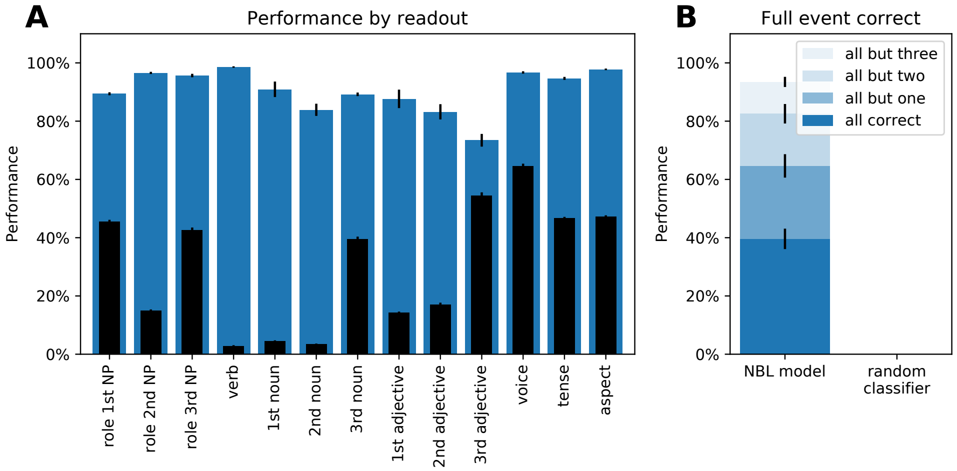


Figure 6.3.: Parallel readout of binding information. A: Performance of the NBL model and the random classifier (black bars) for each of the parallel readouts separately. **B:** Dark blue: performance when counting only the instances where all readouts were correct. Lighter blues: performance when allowing an incorrect assignment in one, two, or three of the thirteen readouts. Note that the random classifier did not satisfy these conditions even a single time for ≈37 000 sentences tested in the twelve different model subjects. Error bars show 95% confidence intervals.

corresponding category names (see section 2.3), and grouped the pronouns. The LIVING/OBJECT category was a set of nouns that could function as grammatical subjects and objects and hence could take on a variety of semantic roles. For each individual noun, the majority of binding queries were correct (at least 51%). The category with the highest binding performance was INSTRUMENT (>99%), while the categories with the lowest performance were the LIVING/OBJECT nouns (61 to 65%) and the pronouns (51%, 54%, 82% and 61% for *he*, *she*, *it* and *they*, respectively). Performance for other categories by word were 77–85% (LIVING), 90–93% (OBJECT) and 91–97% (LOCATION). I found that most confusions occurred within category, in particular, within the LIVING/OBJECT category (see darker off-diagonal color in figure 6.2). Pronouns could occur in all constructions, assume different positions, and many syntactic and semantic roles, which caused the most confusion with other noun categories.

6.3.2 Parallel readout of binding information

In this section, I show that readouts can extract several different kinds of information from the same network activity in the NBL model using a set of parallel readouts.

Each of them was trained independently with the respective target output. Beyond reading out semantic role, noun, adjective and verb identity, I also read out the event semantics voice, tense and aspect. This would allow me to reconstruct the event semantics that went into the generation of each sentence. The performances on this task are displayed in Figure 6.3. As reference model, I again used a random classifier. Note that in panel A, the performance of the random classifier differs greatly between readouts: from 3±0% for the verb to 60±1% for the voice readout. This is because the number of options and the distribution of these options differ between readouts (see section 6.2.2). There are only two voices, active and passive, but there are 48 verbs in the language, which makes guessing the verb identity less likely to be correct. The model performance varies less and is always above 73%. When comparing each of the readouts individually to the random classifier, I find that the NBL model always performs better. Using paired-samples t-tests, I find significant differences between the model and the random classifier for each readout (for each $t(11)>18$, $p<0.001$). To assess how well the information conveyed by each sentence as a whole was represented, I combined all readouts and evaluated in how many cases the NBL model was able to read out every single property of the given sentence correctly (Figure 6.3B). I found that all readouts made the correct choice in 40±5% of the cases. If I allowed one mistake, performance in the NBL modelincreased to 65±6%. Allowing another mistake, the performance was 83±5%, and 93±3% for a third one. By comparison, the random classifier performed at 0±0% even when allowing three mistakes. This suggests that detailed information about a processed sentence is represented in the network and can be extracted reliably, including the semantic roles of the various noun phrases, word identities, and event semantics. When combined, these slot/filler binding relations constitute a sentence-level interpretation.

6.3.3 Problem of 2 sentences

Here I tested whether the NBL model could identify different instances of the same noun within the same sentence. I used four constructions designed to generate Po2 sentences with a probability of 12.5% each. In these sentences, the reoccurring nouns were always in the first and second noun phrases. Using the data on these sentences from the last section, I tested if (i) both nouns could be identified as being the same, using readouts for first and second noun identity; and (ii) if both of the adjectives corresponding to the nouns were identified correctly, using readouts for first and second adjective identity. Figure 6.4A shows that both nouns were correctly identified in 98±1% of all Po2 sentences and both adjectives were correctly identified in 94±3%

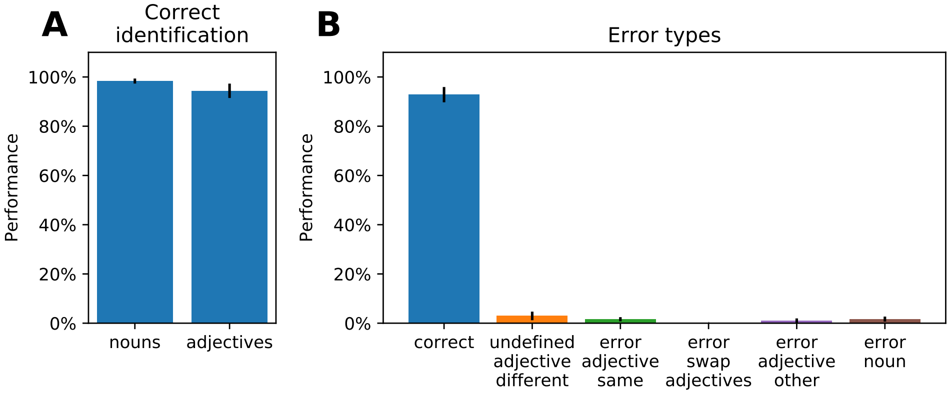


Figure 6.4.: Correct assignments and error types in problem-of-2 sentences. A: Performance in identifying both nouns or both adjectives. **B:** Distinction of different error types, examples for each type are shown in Table 6.4. Error bars show 95% confidence intervals.

of the cases. Furthermore, I distinguished five different error types to determine which kinds of errors occurred. A relevant question here is how often the adjective for the first noun was attributed to the second one, and vice versa. These five error types can be sorted into three groups of errors: First, it is possible that the two instances of the noun cannot be distinguished (noun identity confusing error, Table 6.4). This means that either both noun instances are bound to the *same adjective* or there are *swapped adjectives* between both noun instances. These are the types of errors one would expect from a connectionist model where two bindings interfere with each other. Secondly, it is possible that information that is not provided by a sentence is accidentally attributed but in a way that preserves the identity of both noun instances (noun identity preserving error). If, for example, the first noun does not have a modifying adjective but the readout assigns an adjective distinct from the one for the second noun, this would be an error that would *not* indicate a failed separation of both instances since they still can be distinguished from each other; see the example in Table 6.4 under error type *undefined adjective different*. And third, it is possible that the readout is erroneous in a way that has no relation to other parts of the sentence (other errors). If one or both of the nouns were read out incorrectly I call this an *error noun*, and if one or both adjectives were read out incorrectly I refer to it as *error adjective*. In these cases, the error is due to spurious activity that causes an incorrect readout and it can not be determined whether the two instances of the noun could still be distinguished. I found that in 93±3% of the tested items, both nouns and adjectives were correctly identified, and hence both instances of the nouns can be distinguished

error group	**error type**	**example readouts**	1st adj.	1st noun	2nd adj.	2nd noun
noun identity confusion errors	error adjective same		*A little cat chases the big cat.*			
		target	little	cat	big	cat
		erroneous	big	cat	big	cat
	error swap adjectives		*A little cat chases the big cat.*			
		target	little	cat	big	cat
		erroneous	big	cat	little	cat
identity preserving error	undefined adjective different		*A cat chases the big cat.*			
		target	NONE	cat	big	cat
		erroneous	little	cat	big	cat
other errors	error adjective other		*A little cat chases the big cat.*			
		target	little	cat	big	cat
		erroneous	little	cat	old	cat
	error noun		*A little cat chases the big cat.*			
		target	little	cat	big	cat
		erroneous	little	cat	big	elephant

Table 6.4.: Examples for error types distinguished in figure 6.4. The five error types are illustrated here with an example sentence as well as the corresponding target and erroneous readouts for the identity of the first and second nouns and adjectives. These types can be assigned to three error groups which categorize the errors in terms of sentence processing mistakes.

successfully (Figure 6.4B). Among the errors, the most frequent one was the noun identity preserving error with 3±2%. The noun confusion errors occurred in 2±1% of all cases, while the *swapped adjectives* error did not occur a single time. Finally, other errors occurred in 2±2% of the cases.

6.4 State-space dynamics

In this section, I explore *how* the NBL model solves the binding problem and the problem of 2. I investigate how the state of the spiking network moves through the state space for different kinds of input. One way to think about spiking networks is as dynamical systems. A *closed* system does not receive input (e.g., information, energy, etc.) and is fully described by the value of all state variables at one point in time plus the "rules" how the system changes over time. The state of the network is the combination of all state variables (here, the membrane potentials, the adaptive neuronal currents, and the synaptic currents) and the rules are the differential equations governing the neuronal and synaptic dynamics (section 2.2). The NBL model, however, is not a closed, or autonomous, dynamical system but an open system that receives input (it is "forced"). Thus, it responds to and processes different inputs. In this section, I show how the state of the system changes due to input words and how a sequence of words is processed. Since the readouts only have access to the membrane potential and the adaptive neuronal currents, I focus on the evolution of these state variables.

To reduce the 4000 dimensions of the state variables to three, I used principal component analysis (PCA) executed on the data from a set of keywords for each plot. The three dimensions shown are the ones that explain the most variance within the data (i.e., the first three principal components) and show the dynamics during a word, as well as the differences between words. I then use these dimensions as coordinate system to also plot data from contextual words to the keywords, so the movement through state space becomes continuous and more intuitively understandable. Below each plot, it is noted which words were the keywords the PCA was done on. Note that state change in the state space does not happen at a constant speed. In particular, the length of a word trajectory in state-space does not indicate the length of the word. Longer words tend to show longer trajectories but (a) different stimulations can cause stronger or weaker state changes and (b) PCA rescales the dimensions according to the variability within them.

Single sentence presentation

Figure 6.5 shows how the state of the network changes as a result of processing a sentence with a repeated noun. We can see a continuous trajectory from the sentence beginning (marked by the cross) to the second instance of the noun *dog* at the end of the sentence. On the Y-Z projection, the two instances of *dog* (both blue) have a similar shape because they were triggered by the same input stimulus. However, it is also visible how the preceding context is integrated with new input. Each word starts where the last one ended and the two instances of *dog*, even though they were delivered by the same input stimulus, are clearly distinct on the X-Z projection. This contextual integration also allows the association of *big* and *little* to the corresponding instance of *dog*.

The way the NBL model uses sequential order to inform binding is reminiscent of dynamic binding (see (4) in section 6.1.1). However, there are no oscillations. The network did not exhibit spontaneous oscillations when stimulated by input and I did not add a mechanism to induce oscillatory behavior. Successful binding in sections 6.3.1 and 6.3.2 shows that oscillations were not necessary to represent binding relations. Importantly, the binding readouts did not have access to the whole

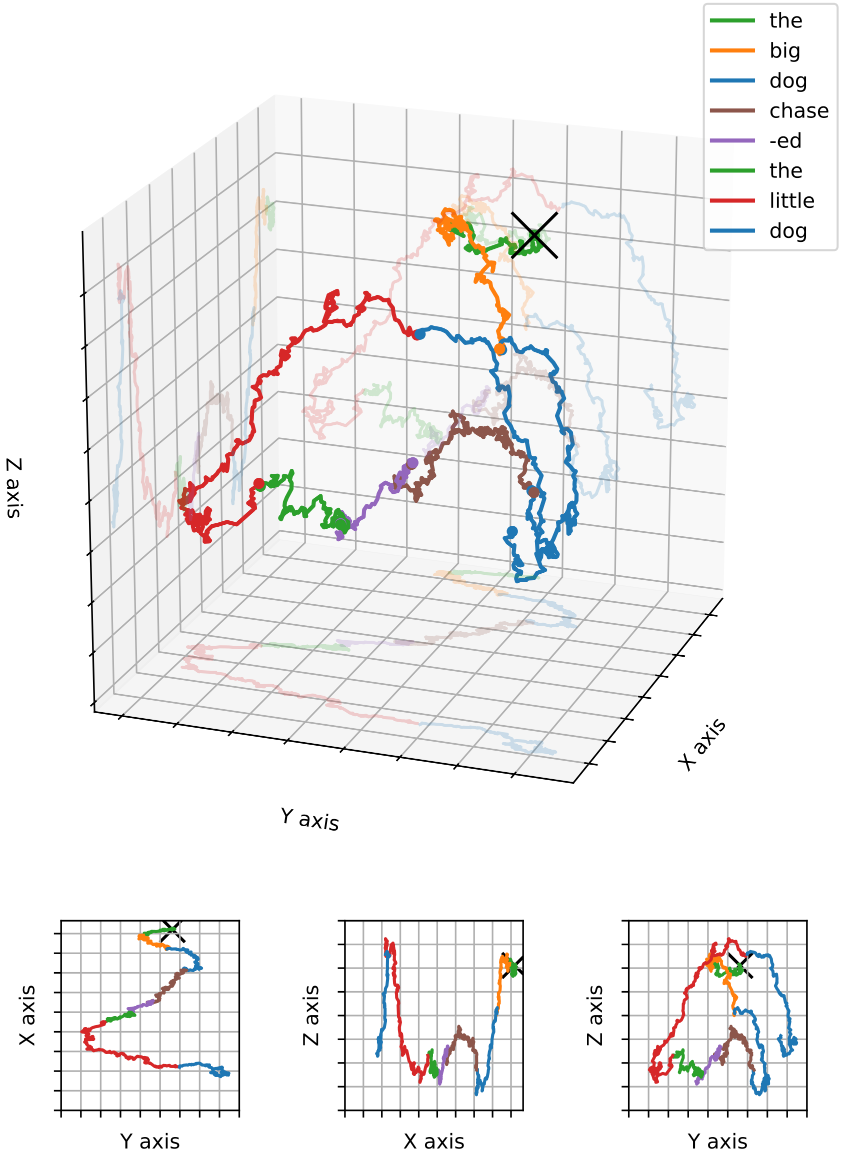


Figure 6.5.: State-space trajectory during a Po2 sentence. Illustration of the example sentence *The big dog chase -ed the little dog*. PCA was run on keywords *big*, *dog*, *little*, *dog*. The cross is marking the sentence start, the trajectory during each word is marked with a distinct color. The top panel shows the 3D plot of the network state with the projections on the respective axial planes plotted faintly. Bottom panels show the same projections in separate plots for clarity. The repeated instance of *dog* is similar in the Y-Z projection but spatially distinct in the X-Z and the X-Y projections.

trajectory, as plotted here, but only to the network state after the last word of each sentence. This means the information of the repeated noun and which adjective has to be bound to which instance of the noun is still available in different dimensions of the network state after the sentence is complete. Previous input words do not need to be reactivated but instead the information remains available within the final state of the network. I showed in chapter 4 how it was possible to store information, such as binding relations, in the activity-silent state variables with long time constants.

Non-linear integration of stimuli

In Figure 6.6, the left panels show three instances where the network with the same initial state (marked by the cross) was stimulated with different adjective-noun pairs. The three adjectives were different from each other while the subsequent noun was the same. The Y-Z projection shows that the trajectories move in three distinct directions,separated by roughly 120 degrees. This is a consequence of the dimensions chosen by the PCA. As before, it can be seen that the position at the end of each noun depends on the preceding adjective, hence the information which adjective was presented before the noun is retained in the end state of each trajectory. However, the combination of adjective/noun input pairs is not purely linear. The three trajectories for *dog* are not spatially shifted (translated) copies of each other but they interact in a more complex fashion with the preceding input. This shows how contextual information was maintained in processing memory.

While the three trajectories for *dog* are roughly parallel to each other on the X-Y projection, they have different directions on the X-Z projection. This multidimensional representation allows the network to store "sameness" between words and as well as the contextual differences at the same time and it uses different dimensions in the state space to make this distinction. This is similar to the conjunctive representations (described in (3) in section 6.1.1). These representations used some form of vector multiplication (e.g., the tensor product or circular convolution) such that the combination of two concept vectors was not just a summation of states but an operation that allowed a more complex interaction between the concepts (Hummel, 2011). Multiplicative interactions like the tensor product have the important property that they are order-sensitive (non-commutative) (Smolensky, 1990). I found that the way the NBL model combines words is also order-sensitive. This can be seen in Figure 6.6, right panels, where I compare a noun preceded by two adjectives with the same noun preceded by the same two adjectives in reverse order. (This is otherwise not possible in the language but was separately simulated for this plot.) Clearly, the

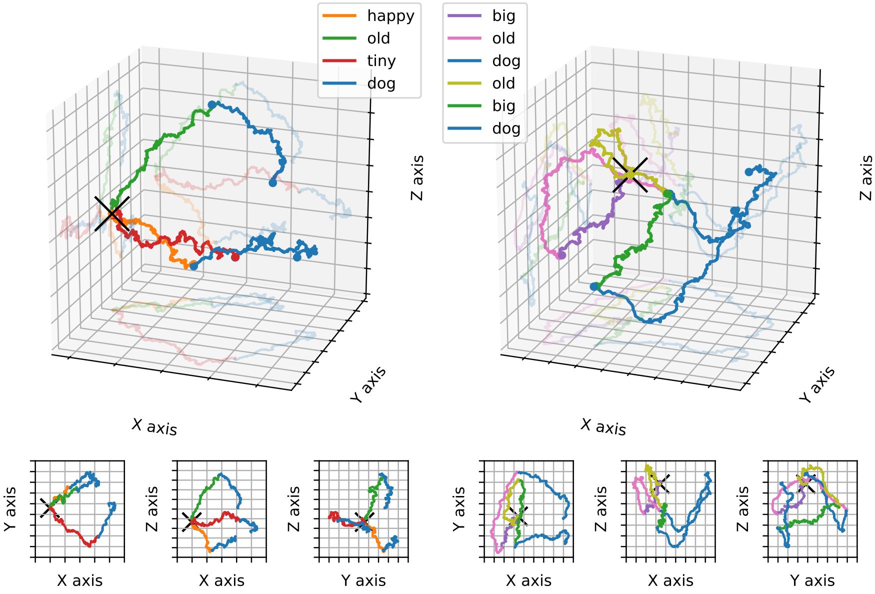


Figure 6.6.: State-space trajectories for different adjective-noun groups. Bottom panels show the axial projections from the larger 3D plots. **Left panels:** Illustration how the network responds to a noun depending on the preceding adjective. The three trajectories for *happy dog*, *old dog* and *tiny dog* all originate at the cross and have the same input stimulation for *dog*. However, the trajectories of *dog* start at different positions and themselves depend on the preceding input word (see X-Z projection). PCA was run on the six keywords in the plot legends. **Right panels:** Illustration of trajectories for *big old* (purple colors) and for *old big* (green colors) followed by the same *dog* stimulus (blue). They originate at the same point marked by the cross. For clarity, two instances of the same adjective have different colors here but the stimulation remained the same. The difference in trajectory and endpoint of both *dog* instances (X-Y projection) shows that context integration is order-sensitive. PCA was run over the first old and both dog instances as keywords.

endpoints of the two trajectories on *dog* are distinct, which is most pronounced in the X-Y projection. This differs from previous proposals for generating conjunctive representations in that no special binding mechanism is required (e.g., tensors). Moreover, the integration process where different features are combined is non-linear because the network is a highly non-linear dynamical system.

Query readout

To illustrate how the query task tested in section 6.3.1 is solved, and how the query is integrated into the network state, in Figure 6.7 I show two queries for three similar sentences in two different coordinate systems (left and right panels) on the same data. The three sentences were of the form *The big dog chases the little cat* but differed in

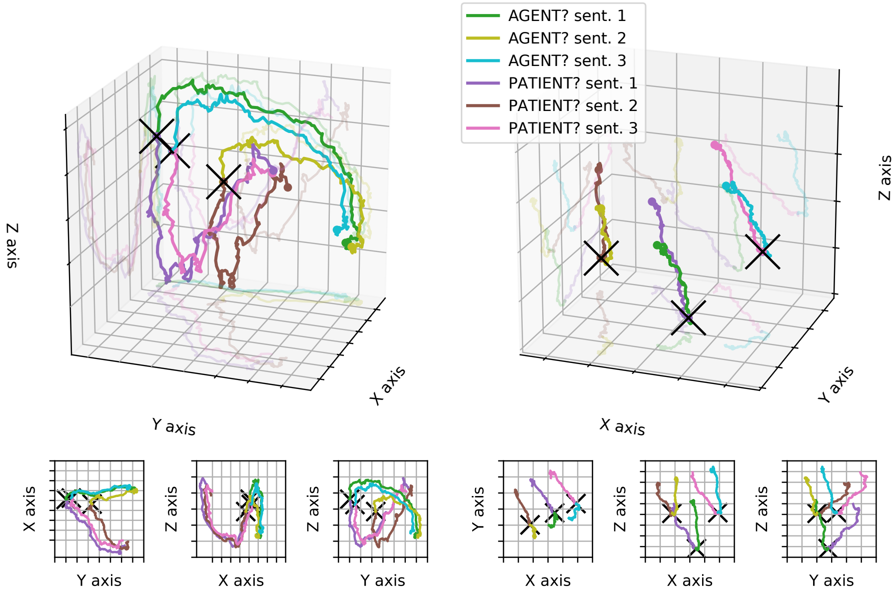


Figure 6.7.: State-space trajectories during the sentence query. The left and the right panels show the same data in two different coordinate systems. Bottom panels show the axial projections from the 3D plot above them. Two queries "AGENT?" and "PATIENT?" are shown for three similar sentences. The endpoints of the sentences, which are the starting points of the query trajectories, are marked by crosses. In the left panels, the PCA was run with the "PATIENT?" and "AGENT?" queries of the first sentence as keywords. In the right panels, the PCA was done with all six queries as keywords. The two plots show that querying the NBL model causes a movement to a query-specific region of state space (left panels). At the same time the network state retains sentence-specific lexical information in other dimensions which the readout can use to answer the query (right panels).

either the second noun or the second adjective. Thus, the network state at the end of each sentence was different. These endpoints are marked by three crosses and are the starting point from where the different query trajectories originate. Since the network is reset to the sentence-final state before querying, both queries to each sentence originate at the same point. By selecting the keywords that the PCA was executed on, the coordinate system in the left panels was chosen to emphasize the state-space movement due to the queries themselves. We see that each "AGENT?" query moves the network state into one region of state space while each "PATIENT?" query moves it to a different region which is most pronounced in the X-Y projection panel. This shows that the queried role itself can be identified from the trajectory endpoints. The coordinate system in the right panels was chosen to emphasize the difference between the final states of the sentences, which is also most distinct in the X-Y projection. Despite the strong forcing through state space due to the query, the sentence-specific lexical content is retained until the end of sentences, and this provides the readout with the information required to answer the query.

6.5 Discussion

In this chapter, I investigated how the NBL model deals with the binding problem in language processing. First, I have shown that it can accomplish explicit binding by identifying the correct noun when queried for a semantic role after sentence presentation. Second, using parallel readouts, I have shown that it can provide information necessary to address the massiveness of the binding problem. The network state contains information about syntactic and semantic features after the complete sentence has been processed and this information can be decoded with linear readouts and transferred to downstream networks for further processing. And third, I have shown that this information was sufficient to solve the multiple instantiation problem, or problem of 2. I found that the NBL model could assign the correct adjective to each instance of a repeated noun in more than 90% of the tested sentences. Finally, by visualizing state-space trajectories, I have shown how the model performed these tasks and that it implements aspects of two connectionist proposals to solve the binding problem in a neurobiologically plausible manner. These results suggest that the NBL model can use a fixed set of primitives and combine them flexibly to represent composed concepts. Furthermore, binding relations are not limited to word pairs (e.g., adjective-noun) but words can be the filler of a variable slot, such as a semantic role. The fact that these tasks were solved with high performance shows that the NBL model implements a mechanism that is suitable for solving binding problems.

By probing state-space trajectories, I could identify similarities between the binding mechanism implemented in the NBL model and previous proposals for binding in connectionist models. The NBL model reflects aspects of two proposals discussed in the introduction: (3) conjunctive coding and (4) dynamic binding. Conjunctive coding is implemented as non-linear integration of primitives: the network state after presenting *black cat* follows from a nonlinear, order-sensitive combination of *black* and *cat*. Since the network has high dimensionality, there are also dimensions in which *cat* can be identified independently of the preceding adjective. However, in contrast to other proposals on how to achieve conjunctive coding (Smolensky, 1990; Gayler, 2004), there is no specialized operation required to achieve this, such as tensor products or circular convolution. At present, it is not known how such mathematical operations on high-dimensional vector representations could be implemented within the brain's neurobiological infrastructure. In contrast, conjunctive coding in the NBL model relies on the nonlinear nature of the network dynamics and its specific connectivity to combine primitives into composed concepts where history-dependent neuronal processes with long time constants (such as spike-rate adaptation) serve to integrate consecutive pieces of information. This neuronal processing memory mechanism has been demonstrated to be sensitive to context of occurrence and serial order relations (Fitz et al., 2020).

Additionally, the NBL model implements aspects of dynamic binding by continuously integrating a stream of information with the preceding context. In figure 6.5, it was shown that the temporal difference between the two presentations of *dog* was reflected in their position in state space. However, in contrast to dynamic binding through synchrony models relying on oscillations, repeated reactivation of representations was not necessary, because the binding information was already maintained in the network state and could be read out at the end of the sentence. Again, there were dimensions in the network that retained information related to which nouns were present in the sentence and other dimensions retained information about which adjective modified which noun. Thus, the NBL model offers a solution to two of the main critiques of binding-through-synchrony approaches using oscillations: that it is not clear (i) what causes periodic reactivation and (ii) how the information reflected in synchronous activity is used by downstream regions. In the NBL model, temporal information can be maintained in the network through neuronal memory mechanisms without oscillations, and downstream regions could, in principle, linearly extract binding relations for further processing, or explicit reasoning about the message of an utterance. The NBL model implements aspects of the dynamic binding through polysynchrony model. This makes sense, since it too is based on spiking neural networks with time-delayed communication. A difference is that in the NBL model, the binding information is stored within the whole neuronal state while in the polysynchrony model, only the elicited spikes carry information. The former has the advantage of building a stable representation that can be read out over a longer time. Otherwise, reading out the spike pattern at the moment of generation is necessary.

This discussion touches upon a conundrum related to the binding problem in connectionist models: On the one hand, stable representations of binding relations are needed to maintain information over long periods of time. On the other hand, timing and the order of input often carry relevant information in language which means that binding also needs to be sensitive to information on shorter timescales. The binding-through-synchrony model addresses this issue by replaying previous information over and over. The NBL model offers an alternative that relies on complex interactions between dynamic variables with long and short timescales. In chapter 5, I showed that readouts can be calibrated to find stable representations of the different semantic roles over the course of a sentence and these were supported by neuronal and synaptic time constants of up to 500 ms. Dynamic variables in biological circuits can act on even longer timescales, for example, in long-term synaptic dynamics, protein synthesis and switching (K. C. Martin & Kosik, 2002; Fallon & Taylor, 2013), and these could be utilized in long-term storage. In this chapter, I showed that stable representations did not interfere with timing information since two instances of the same word could be distinguished and independently identified at the end of a sentence. Thus, the NBL model provides both temporally stable representations while also being sensitive to timing information.

In section 6.1.2, I described three ways to implement a solution to the problem of 2 (Sougné, 2006): (1) load several instances of the repeated noun into working memory, (2) rely on multiple instances of the same noun in long-term memory and (3) repeatedly activate the same noun during each period of an oscillation when using dynamic binding-through-synchrony. Of these proposals, (1) is the closest approximation to the mechanism implemented in the NBL model since there is only one instance per noun in long-term memory, excluding (2), and the NBL model does not exhibit oscillations, excluding (3). However, there are no static memory registers in the NBL model. As discussed in chapter 4, processing memory in the model is based on neuronal and synaptic processes with long time constants. The information held in these dynamic variables constantly changes and continuously influences the network state. Thus, dynamic memory registers in the NBL model actively transform information during sequential processing in unification space, the opposite of a static register memory that maintains information passively and unchanged. Therefore, the different instances of the same noun carry a temporal signature in memory and this is what makes them distinguishable for the readout processes. In the model proposed here, variables and fillers are not connected by a physical link that represents binding as, for example, in van der Velde and de Kamps (2006). Rather, binding is implicit in a dynamic processing memory. A value is bound to a variable if the value can be obtained by querying the network with a semantic variable, and vice versa (van der Velde & de Kamps, 2015). If the value of a semantic variable differs between two sentences, this will be reflected in different network trajectories through state space. This is what I demonstrated in the query task. If a noun was the filler to a variable like “agent of the sentence”, processing in the NBL model was influenced such that it was possible to extract the binding relation from a query. This mechanism does not require explicit, physical links between representations that are bound temporarily.

In section 6.4, I showed that there was a dimension in state space that encoded the identity of nouns in a sentence, a different dimension for the position of the nouns, and yet a different dimension encoding adjectives, and so on. Thus, the NBL model might suffer from an increasing need of necessary resources to deal with different binding relations. In fact, in this chapter I had to increase network size and reduce lexicon size to obtain high performance on the query task. The number of different binding relations that the network needs to be able to distinguish can be estimated as the number of constructions in the language (12), times the number of queries (7), times the number of possible nouns (40) which yields 3360 different combinations, and this is close to the number of network dimensions used here (4000 neurons). However, readouts can linearly combine any subset and number of network dimensions and project them onto a target output to solve simple tasks. When using many parallel readouts, which does not change the network dimensions, these can be combined to allow solving more complex tasks. For example, in section 6.3.2, I read out information related to thirteen different aspects of the sentence content. Multiplying the number of different readout states yields $\sim 10^{12}$ potential configurations and the NBL model was able to solve this task in more than 90% of the cases using just 4000 dimensions. This is because the response of each of these readouts can be combined differently depending on the task requirements, and this is similar to choosing a different coordinate system to visualize state-space trajectories in section 6.4. Note that in a setup with a more complex readout, one that does not rely on linear separability, more information can be extracted from individual dimensions in state space. In chapter 3, I showed that the NBL model can separate spike information that is not linearly separable in the dimensions of the input channels. This allows denser information transfer to downstream networks than the linear readouts in the NBL model.

In this chapter, I have illustrated the value of neurobiological modeling in the context of the binding problem in language processing. We know that the brain can perform linguistic binding operations, and we also know that the theory of neural networks is rich enough to capture classical computability theory (Siegelmann, 1999) where binding is achieved through static memory registers (variables/slots) that store numbers (values/fillers). Hence, the most relevant, open question is how this is accomplished within neurobiological infrastructure. The account presented here argues that this feat could be implemented through physiological variables that control neuronal adaptation and provide dynamic memory registers for temporary binding on short timescales (see also Fitz et al., 2020). Since artificial neural networks do not typically use neurons with memory and history-dependent behavior, they might be unable to solve this problem even in principle. This is an important contribution to an ongoing discussion and it demonstrates the potential of the approach taken in this dissertation.

7

General discussion

In this dissertation, I presented the notion of a neurobiological model and investigated the properties of one specific instance, the NBL model. On this instance, I showed how to realize some of the benefits of neurobiological models as outlined in section 1.1.

The NBL model reflects a number of neurobiological features that previous models of sentence processing do not have. It uses a network composed of neurons with intrinsic plasticity that communicate through spikes which have a lasting effect on post-synaptic targets. In chapter 2, I outlined how these model features capture properties of biology at the level of neurons and small networks. These properties also reflect biophysical constraints that are present in the brain and limit the processing mechanisms (i.e., algorithms) that can be realized in neural infrastructure. Implementing these constraints in the NBL model results in a sentence processing model with many independent parameters (see section 2.2), however, the values of these parameters are constrained by the physiology of biological neurons and networks. Within those limits, I explored different model characteristics and investigated their effect specifically on the processing capacity of the NBL model.

In chapter 3, I compared different ways to inject input into the NBL model and investigated how this encoding influenced information processing. I compared three different encoding schemes; a spike-timing based temporal encoding, a rate encoding and an encoding based on direct current stimulation. I found that the different encodings had a strong influence on network processing. Using benchmark tasks, I investigated how well the model separated input stimuli, retained them in processing memory and integrated adjacent items in sequences. The results showed that temporal and direct current encoding was superior to rate encoding in terms of accuracy on these tasks. The former two encodings performed similarly well. I tested the interaction of these encoding schemes with the type of input projection to the network. In the stimulus-specific projection a dedicated subset of neurons was targeted by each stimulus, while the stimulus-general projection targeted the whole network. Although all encodings benefited from using a specific projection, I found that the performance of the rate encoding dropped more than the performance of the other two encodings when using a general projection. Temporal encoding allowed the separation of a large number of different stimuli even though it used different instances of the same random Poisson process to identify distinct stimuli. With the parameters I used in the simulations, I could distinguish up to one stimulus for every two neurons in the network. Finally, I tested to what extent noise affected encoding and found that all three encoding schemes showed high noise tolerance with a stimulus-specific projection. With the general projection, temporal encoding performed best.

These results show that the spike-based temporal encoding was robust to noise and under some circumstances it performed equally well or better than direct current encoding. Neuronal stimulation with direct currents is not a biologically plausible mode of communication between neurons or brain regions but only served as an idealized benchmark. Furthermore, the results show that one population of neurons can encode a significant number of different stimuli without exhibiting a changing firing rate or using a spatial component in encoding.

I further investigated the relationship between model parameters and processing characteristics in chapter 4. I compared recurrent and feed-forward networks with different connection densities and found that the processing memory in the NBL model was not enhanced by recurrent connectivity. Furthermore, I compared networks with and without spike rate adaptation, a form of intrinsic neuronal plasticity, and with different synaptic current time constants. I found that both intrinsic plasticity and long synaptic currents supported memory. Without these features, there was insufficient processing memory. These memory mechanisms did not rely on ongoing spiking interaction during memory maintenance and thus served as an activity-silent processing memory with lower metabolic cost (as introduced in Stokes, 2015).

These results establish a causal relationship between neurobiological features of the modeled system and its processing capacity. They identify neuronal and synaptic processes with long time constants as instrumental for processing memory. Because the constraints of the NBL model reflect neurobiological constraints, this suggests that this causal relationship also holds in real biological networks (Titley et al., 2017). Such an activity-silent dynamic memory mechanism is an alternative to static memory registers. It enables continuous reading and writing between coupled dynamical processes and through these interactions implements a basic read-write memory system (Petersson et al., 2008). Processes that run on longer time scales, such as intrinsic neuronal plasticity and slow synaptic currents, act as memory variables while processes with short time constants can function as writing operations that are processing information simultaneously. At the same time, the dynamic variables serving as memory continuously influence the membrane state and therefore the spiking activity itself. Information is continuously read from the memory variables and is thus folded into active processing. This activity-silent memory architecture relies on the continuous integration of past and new information which is suitable for online, incremental and context-dependent language processing (Marslen-Wilson, 1975; Kutas & Hillyard, 1980).

However, one interesting result in chapter 4 was that I could not find a beneficial effect of recurrent connectivity. This is unlikely to be the case in real biological systems, given the ubiquity of strong recurrent connectivity in the brain (Markram et al., 2015; Izhikevich & Edelman, 2008). This could be due to the nature of the random connectivity in the network of the NBL model. It is conceivable that recurrent connectivity is beneficial if it is more specific and targeted, as it is in cortical connectivity (Douglas et al., 1995), or in the context of task-relevant learning. This mismatch between model behavior and experimental evidence invites additional research using neurobiological models with more structured connectivity, as I will discuss below.

The investigations in chapters 3 and 4 concluded the testing of different parameter choices in the NBL model. As stated in the introduction, the more similar a neurobiological model is to the cortical infrastructure for language, the more it will reflect human processing characteristics. The question I addressed in the subsequent chapters was whether the model was sufficiently realistic to reproduce effects that have been observed in human language processing.

In chapter 5, I tested whether I could find similarities between the processing in the NBL model and human sentence comprehension. I found that the NBL model does exhibit several human processing characteristics when performing an online shallow semantic analysis of sentences generated by an English-like language. It showed contextual, incremental and predictive processing, and was able to integrate both syntactic and semantic constraints concurrently. I also found that the model could generalize from a small training set to novel sentences and unfamiliar nouns by inferring semantic properties of novel words from context.

These findings suggest that our current scientific understanding of biological networks is sufficient to reproduce basic aspects of language processing. Rather than being baked into the model design, these processing characteristics emerge from modeling the underlying neurobiological infrastructure. Some of these processing effects have previously been demonstrated in connectionist models, e.g., the contextual integration of words in semantic processing (St. John & McClelland, 1990). However, these models are much less biologically plausible compared to the NBL model. The results also suggest that it is beneficial to use biophysical model components where dynamic variables have physical units of measurement. For instance, simulations of the NBL model are running in continuous physical time rather than discrete time steps, or abstract model time. Thus, in contrast to standard cognitive models, time in the NBL model can be systematically related to time and timing in human language processing, both at the level of behavior but also physiology. As a case in point, words were presented with a duration of 50 to 500 ms, giving rise to sentences with a duration of several seconds of real time. I found that neurobiological processes acting on comparable time scales were required in the model to achieve sentence comprehension on behavioral time scales. This suggests that similar neurobiological processes in the brain might be causally relevant for supporting sentence processing at different temporal grain sizes (Hasson et al., 2015).

Finally, in chapter 6, I used the NBL model to identify a neurobiological mechanism suitable for addressing the binding problem. I investigated whether the model could establish binding relations and analyzed how it represented this information. I tested binding using a question-answering protocol. After presenting a sentence, I queried the network with a semantic role variable (e.g., AGENT or PATIENT of an action) and found that nouns that were temporarily bound to these roles could be decoded from the network activity with high performance. Using parallel readouts, I also demonstrated that the network maintained critical information concurrently, such as noun and verb identities, adjectives, and the event semantics of sentences. This information was sufficient to resolve binding relations in sentences with repeated occurrences of the same noun – so-called problem-of-2 sentences. To investigate how the NBL model solved binding, I analyzed the trajectories of the network through its high-dimensional state space. I found that information was stored in the neuronal memory variables of the NBL model which influenced ongoing processing and implicitly represented binding relations that could be extracted by a downstream readout if needed.

In addition, these results also show how models that implement biophysical constraints can help to address issues occurring in models that only respect some or no biophysical constraints. If a model encounters processing limitations, such as the binding problem, this can either reflect limitations of the underlying biological system or it can reflect limitations introduced by the specific properties of the model. Models that implement biophysical constraints can help to distinguish these two cases. Here I found that, while neurons without internal dynamics may not be able to represent binding information, more complex neurons, closer to biophysical reality, are able to do so.

To summarize, the insights gained in this dissertation are three-fold. Firstly, the NBL model is suitable to connect detailed neurobiology to cognitively relevant language processing tasks. This means that the scientific understanding of neurobiology is sophisticated enough to build models that bridge to a cognitive level. Secondly, I found that processing limitations imposed by the neurobiological constraints of the model are different than may be naively assumed. The encoding with spike patterns without spatial or rate encoding proved to be powerful enough to discern a large number of stimuli. Generalization in the NBL model was such that relatively little training data was sufficient to extrapolate to a larger data set including entirely novel words and contexts. At the same time, the processing memory in the NBL model was limited. Only long neuronal or synaptic time constants could provide sentence-level processing memory, while network size and connectivity contributed relatively little to memory. And thirdly, the neurobiological properties of the model provided insights into possible implementations of processing aspects in biological neural networks. Processing memory could be provided by neuronal adaptation and synaptic currents rather than recurrent connectivity. The binding problem could be addressed through the rich dynamic representation of information that is present within neuronal memory.

Outlook

I see the work presented in this dissertation as a fruitful initial step of a research trajectory that aims to exploit the benefits of neurobiological models to improve our understanding of sentence processing in the brain. Extending this research will involve incorporating additional aspects of neurobiology to reflect more constraints of the neuronal infrastructure in these models.

One possible next step is to use more detailed neurobiological data including different electrophysiological neuron classes, connectivity data, and synaptic heterogeneity. As described in chapter 1, neurons in the brain exhibit different electrophysiological behaviors that define different response types. Neurons of each class have distinct spatial distributions in the cortical layers and distinct connection probabilities depending on the electrophysiological neuron type they are connecting to (Izhikevich & Edelman, 2008). Furthermore, the type of synapse connecting two neurons depends on the neurons that are involved (Cauli et al., 1997; Markram et al., 2004). The relevant data is available in the form of large-scale connectomes (e.g., Markram et al., 2015). Using such a detailed model of neurobiology also invites the use of heterogeneous parameters within each neuron population. In the simulations conducted in this dissertation, all neurons were characterized by the same neuronal parameters. In cortex, neurons belong to different neuronal classes, which can be formalized by different sets of neuronal parameters, and these parameters vary within one class as well (Markram et al., 2015). Simulation studies indicate that this heterogeneity can have an important functional role, such as extending the network-internal time scales, and therefore processing memory of the network (Duarte & Morrison, 2019).

In combination with further neuronal and synaptic processes that implement learning and development, such as STDP (Markram et al., 1997; Kempter et al., 1999), these more detailed biophysical models may be a suitable substrate to address the functional role of recurrent connectivity in neural networks, a question that could not be satisfyingly addressed in this dissertation. Similarly, modeling different neuron classes allows for the investigation of their functional role (Wang et al., 2004; Haeusler & Maass, 2007). Results in chapter 4 suggest that neurons with adaptive electrophysiological properties are important for processing memory. Non-adapting or bursting neurons might be important for other processing aspects. Okun et al. (2015) identified “soloists” and “choristers” among the spiking behavior of a population of neurons in mouse cortex, describing how much the activity of a neuron correlates with the activity of other neurons of the same population. It is possible that such a distinction emerges in neurobiologically more detailed models (Markram et al., 2015). Modeling different electrophysiological neuron types may also facilitate the bridging of neurobiological models to neuroimaging data such as local field potentials (LFPs). LFPs are measured in EEG or MEG experiments and are mainly generated by synaptic currents, in particular in excitatory pyramidal neurons (Nunez & Srinivasan, 2006). There are accurate physical models that generate EEG data from simulated network activity (Mazzoni et al., 2015; Nunez & Srinivasan, 2006) which can then be related to data from neuroimaging experiments and generate novel predicitions.

A further extension of this research trajectory could be to combine neurobiological models with *in vitro* models of neuron populations to investigate processing properties of biological networks and how they are influenced by mediating factors, such as genes and proteins. Genes have been indicated as mediators of the development and function of networks involved in language processing and have been shown to affect language abilities (Enard et al., 2002; Folia et al., 2011). Using neurobiological models in combination with *in vitro* models allows for the investigation of how genes affect processing at the level of neural networks. This could be done by using human stem cells to grow populations of neurons (Günlahar & Kuschner, 2016). The properties of these populations can then be recorded through synapse staining (Ippolito & Eroglu, 2010) and calcium imaging of network activity (Chen et al., 2013) which can be used to set the parameters in a neurobiological model of sentence processing. Using this *in silico* model, structured sequence processing tasks can be investigated with respect to the underlying genes that influenced the measured neuronal and synaptic parameters. Comparing *in silico* models of genetically mutated neuron populations (for example CNTNAP2 knock-outs, Folia et al., 2011) with the *in silico* models of control populations can then identify the functional effect of a language-related genetic mutation on the processing characteristics of the network.

Apart from these possibilities, neurobiological models can help to focus and advance cognitive modeling efforts. If models of cognitive processes do not capture experimentally measured behavior, this is due to abstractions made from biology, suggesting that important features of the real system that generated this behavior are missing. One such abstraction, for example, is the use of symbolic computations and production rules where it is unclear how these mechanisms relate to the underlying system. Neurobiological models also use abstractions from biophysical reality but the relationship to the real biological system is more transparent. For instance, a point neuron is abstracting away the dendritic tree but still does spike-based processing. But if there are good reasons to believe that non-linear integration in dendrites matters functionally, the neuron models can be augmented accordingly. Hence, these abstractions can be investigated systematically to determine the cause for a discrepancy between model behavior and experimentally observed behavior. If an abstraction is identified as critical, neurobiology itself determines how to extend the model in order to address this difference. This is in contrast to models that do not implement biophysical constraints which can be extended arbitrarily. At some point, neurobiological models will be detailed enough to capture most relevant properties of the language system because they are sufficiently close to the neural infrastructure responsible for language processing.

Moreover, using biophysical models can facilitate scientific collaboration since these models provide a common framework for addressing research questions in the language sciences as well as other cognitive domains. Unless there is a substantial revolution within our understanding of the brain's neurobiology, any neurobiological model will reflect the properties I implemented or state how the implementation abstracts away from them. It will reflect that the brain's computation is based on neural networks with spiking, adapting neurons that have fast synapses with extended synaptic currents. Hence, neurobiological models addressing different questions in the language sciences or other cognitive domains will have a common foundation. This allows different disciplines to operate within the same constraint space, sharing a common foundation. Different neurobiological models can naturally be compared and related to each other, or be integrated to explore more complex behaviors. This may allow for a gradual development of more explanatory models of human cognitive capacities.

Final words

In this dissertation, I presented the notion of a neurobiological model and investigated the properties of one specific instance, the NBL model. The insights gained from this investigation illustrate the potential of this line of scientific research. I was able to simulate a neural network based on insights from experimental data at the neuronal level that could be used in the NBL model to solve cognitively relevant tasks in language processing, such as semantic role labeling and binding query resolution. I showed that neurobiological models can be used to establish causal relationships between biophysical properties reflected in the model and the processing properties it exhibits. Respecting the constraints on processing exerted by the properties of the biological system can solve questions raised in models that implement less biophysical constraints.

These results show that it is possible to harness detailed experimental insights on the level of neurons and networks to improve our understanding of cortical processes. Ultimately, the investigation and development of neurobiological models will provide a deeper mechanistic understanding of cognitive processes based on the behavior and interactions of neurons in networks.

Bibliography

Adrian, E. D. (1926). The impulses produced by sensory nerve endings. *The Journal of Physiology*, *61*(1), 49-72.

Ahissar, E., Sosnik, R., & Haidarliu, S. (2000). Transformation from temporal to rate coding in a somatosensory thalamocortical pathway. *Nature*, *406*(6793), 302-6.

Altmann, G. T. (1995). *Cognitive models of speech processing: Psycholinguistic and computational perspectives*. MIT Press.

Altmann, G. T., & Kamide, Y. (1999). Incremental interpretation at verbs: Restricting the domain of subsequent reference. *Cognition*, *73*(3), 247–264.

Altmann, G. T., & Mirković, J. (2009). Incrementality and prediction in human sentence processing. *Cognitive Science*, *33*(4), 583-609.

Amit, D. J., & Mongillo, G. (2003). Selective delay activity in the cortex: Phenomena and interpretation. *Cerebral Cortex*, *13*(11), 1139–1150.

Arnon, I., & Snider, N. (2010). More than words: Frequency effects for multi-word phrases. *Journal of Memory and Language*, *62*(1), 67-82.

Barak, O., & Tsodyks, M. (2014). Working models of working memory. *Current Opinion in Neurobiology*, *25*, 20–24.

Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, *68*(3), 255–278.

Bartol, T. M., Bromer, C., Kinney, J., Chirillo, M. A., Bourne, J. N., Harris, K. M., & Sejnowski, T. J. (2015). Nanoconnectomic upper bound on the variability of synaptic plasticity. *eLife*, *4*, e10778.

Berko, J. (1958). The child's learning of English morphology. *Word*, *14*(2-3), 150–177.

Bernacchia, A., Seo, H., Lee, D., & Wang, X.-J. (2011). A reservoir of time constants for memory traces in cortical neurons. *Nature Neuroscience*, *14*, 366–372.

Bever, T. G. (1970). The cognitive basis for linguistic structures. *Cognition and the Development of Language*, *279*(362), 1–61.

Boerlin, M., Machens, C. K., & Denève, S. (2013). Predictive coding of dynamical variables in balanced spiking networks. *PLOS Computational Biology*, *9*(11), e1003258.

Bono, J., & Clopath, C. (2017). Modeling somatic and dendritic spike mediated plasticity at the single neuron and network level. *Nature Communications*, *8*(1), 706.

Borst, A., & Theunissen, F. E. (1999). Information theory and neural coding. *Nature Neuroscience*, *2*(11), 947–957.

Brette, R. (2015a). Philosophy of the Spike: Rate-Based vs. Spike-Based Theories of the Brain. *Frontiers in Systems Neuroscience*, *9*, 151.

Brette, R. (2015b). What Is the Most Realistic Single-Compartment Model of Spike Initiation? *PLOS Computational Biology*, *11*(4), e1004114.

Brette, R., & Gerstner, W. (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of Neurophysiology*, *94*(5), 3637–3642.

Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J. M., … Destexhe, A. (2007). Simulation of networks of spiking neurons: A review of tools and strategies. *Journal of Computational Neuroscience*, *23*(3), 349–398.

Brodmann, K. (1909). *Vergleichende Lokalisationslehre der Grosshirnrinde in ihren Prinzipien dargestellt auf Grund des Zellenbaues*. Barth.

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., & Mercer, R. L. (1993). The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, *19*(2), 263–311.

Calvo, P., & Symons, J. (2014). *The Architecture of Cognition: Rethinking Fodor and Pylyshyn's Systematicity Challenge*. Cambridge, USA: MIT Press.

Carlson, K. D., Richert, M., Dutt, N., & Krichmar, J. L. (2013). Biologically plausible models of homeostasis and STDP: Stability and learning in spiking neural networks. In *The 2013 International Joint Conference on Neural Networks (IJCNN)* (p. 1-8).

Cauli, B., Audinat, E., Lambolez, B., Angulo, M. C., Ropert, N., Tsuzuki, K., … Rossier, J. (1997). Molecular and Physiological Diversity of Cortical Nonpyramidal Cells. *Journal of Neuroscience*, *17*(10), 3894-3906.

Chambers, C. G., Tanenhaus, M. K., Eberhard, K. M., Filip, H., & Carlson, G. N. (2002). Circumscribing Referential Domains during Real-Time Language Comprehension. *Journal of Memory and Language*, *47*, 30–49.

Chang, F. (2002). Symbolically speaking: A connectionist model of sentence production. *Cognitive Science*, *26*(5), 609-651.

Chang, F. (2009). Learning to order words: A connectionist model of heavy NP shift and accessibility effects in Japanese and English. *Journal of Memory and Language*, *61*(3), 374-397.

Chang, F., Dell, G. S., & Bock, K. (2006). Becoming syntactic. *Psychological Review*, *113*(2), 234.

Chaudhuri, R., & Fiete, I. (2016). Computational principles of memory. *Nature Neuroscience*, *19*(3), 394.

Chechik, G., Meilijson, I., & Ruppin, E. (1998). Synaptic Pruning in Development: A Computational Account. *Neural Computation*, *10*(7), 1759-1777.

Chen, T.-W., Wardill, T. J., Sun, Y., Pulver, S. R., Renninger, S. L., Baohan, A., … Kim, D. S. (2013). Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, *499*(7458), 295-300.

Christiansen, M. H., Allen, J., & Seidenberg, M. S. (1998). Learning to Segment Speech Using Multiple Cues: A Connectionist Model. *Language and Cognitive Processes*, *13*(2-3), 221-268.

Christiansen, M. H., & Chater, N. (1994). Generalization and Connectionist Language Learning. *Mind & Language*, *9*(3), 273-287.

Christiansen, M. H., & Chater, N. (2001). *Connectionist Psycholinguistics*. Greenwood Publishing Group.

Clopath, C., Jolivet, R., Rauch, A., Lüscher, H.-R., & Gerstner, W. (2007). Predicting neuronal activity with simple models of the threshold type: Adaptive Exponential Integrate-and-Fire model with two compartments. *Neurocomputing*, *70*(10), 1668-1673.

Connors, B. W., & Gutnick, M. J. (1990). Intrinsic firing patterns of diverse neocortical neurons. *Trends in Neurosciences*, *13*(3), 99-104.

Csicsvari, J., Hirase, H., Czurko, A., Mamiya, A., & Buzsáki, G. (1999). Fast network oscillations in the hippocampal CA1 region of the behaving rat. *Journal of Neuroscience*, *19*(RC20), 1–4.

Curtis, C. E., & D'Esposito, M. (2003). Persistent activity in the prefrontal cortex during working memory. *Trends in Cognitive Sciences*, *7*(9), 415–423.

Dayan, P., & Abbott, L. F. (2001). *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. Cambridge, USA: MIT Press.

Deco, G., Jirsa, V. K., Robinson, P. A., Breakspear, M., & Friston, K. (2008). The Dynamic Brain: From Spiking Neurons to Neural Masses and Cortical Fields. *PLOS Computational Biology*, *4*(8), e1000092.

Dell, G. S. (1986). A spreading-activation theory of retrieval in sentence production. *Psychological Review*, *93*(3), 283.

DeLong, K. A., Urbach, T. P., & Kutas, M. (2005). Probabilistic word pre-activation during language comprehension inferred from electrical brain activity. *Nature Neuroscience*, *8*(8), 1117-1121.

Demberg, V., & Keller, F. (2019). Cognitive models of syntax and sentence processing. In P. Hagoort (Ed.), *Human Language: From Genes and Brain to Behavior.* Cambridge, MA: MIT Press.

Destexhe, A., Mainen, Z. F., & Sejnowski, T. J. (1994). Synthesis of models for excitable membranes, synaptic transmission and neuromodulation using a common kinetic formalism. *Journal of Computational Neuroscience*, *1*(3), 195-230.

Destexhe, A., & Paré, D. (1999). Impact of network activity on the integrative properties of neocortical pyramidal neurons in vivo. *Journal of Neurophysiology*, *81*(4), 1531–1547.

Douglas, R. J., Koch, C., Mahowald, M., Martin, K. A., & Suarez, H. H. (1995). Recurrent excitation in neocortical circuits. *Science*, *269*(5226), 981-985.

Duarte, R., & Morrison, A. (2014). Dynamic stability of sequential stimulus representations in adapting neuronal networks. *Frontiers in Computational Neuroscience*, *8*, 124.

Duarte, R., & Morrison, A. (2019). Leveraging heterogeneity for neural computation with fading memory in layer 2/3 cortical microcircuits. *PLOS Computational Biology*, *15*(4), e1006781.

Duarte, R., Uhlmann, M., van den Broek, D., Fitz, H., Petersson, K. M., & Morrison, A. (2018). Encoding symbolic sequences with spiking neural reservoirs. In *2018 International Joint Conference on Neural Networks* (pp. 1–8). Rio de Janeiro, Brasil.

Durstewitz, D., Seamans, J. K., & Sejnowski, T. J. (2000). Neurocomputational models of working memory. *Nature Neuroscience*, *3*(11s), 1184.

Dyer, M. G. (2006). Will the neural blackboard architecture scale up to semantics? *Behavioral and Brain Sciences*, *29*(1), 77-78.

Eckhorn, R., Bauer, R., Jordan, W., Brosch, M., Kruse, W., Munk, M., & Reitboeck, H. J. (1988). Coherent oscillations: A mechanism of feature linking in the visual cortex? *Biological Cybernetics*, *60*(2), 121–130.

Eichenbaum, H., & Cohen, N. J. (2004). *From conditioning to conscious recollection: Memory systems of the brain* (No. 35). Oxford University Press.

Eliasmith, C. (2013). *How to build a brain: A neural architecture for biological cognition*. Oxford University Press.

Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., & Rasmussen, D. (2012). A large-scale model of the functioning brain. *Science*, *338*, 1202–1205.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Enard, W., Przeworski, M., Fisher, S. E., Lai, C. S. L., Wiebe, V., Kitano, T., … Pääbo, S. (2002). Molecular evolution of *FOXP2*, a gene involved in speech and language. *Nature*, *418*(6900), 869-872.

Fallon, J. R., & Taylor, A. B. (2013). Protein synthesis in neurons. In *els.* Chichester: American Cancer Society.

Feldman, J. (2013). The neural binding problem(s). *Cognitive Neurodynamics*, *7*(1), 1-11.

Feldmeyer, D., & Sakmann, B. (2000). Synaptic efficacy and reliability of excitatory connections between the principal neurones of the input (layer 4) and output layer (layer 5) of the neocortex. *The Journal of Physiology*, *525*(1), 31-39.

Fitz, H., & Chang, F. (2008). The role of the input in a connectionist model of the accessibility hierarchy in development. In *32nd Annual Boston University Conference on Language Development [BUCLD 32]* (pp. 120–131). Cascadilla Press.

Fitz, H., & Chang, F. (2009). Syntactic generalization in a connectionist model of complex sentence production. In *Connectionist Models of Behaviour and Cognition II* (Vol. Volume 18, p. 289-300). World Scientific.

Fitz, H., & Chang, F. (2017). Meaningful questions: The acquisition of auxiliary inversion in a connectionist model of sentence production. *Cognition*, *166*, 225–250.

Fitz, H., & Chang, F. (2019). Language ERPs reflect learning through prediction error propagation. *Cognitive Psychology*, *111*, 15-52.

Fitz, H., Uhlmann, M., van den Broek, D., Duarte, R., Hagoort, P., & Petersson, K. M. (2020). Neuronal spike-rate adaptation supports working memory in language processing. *Proceedings of the National Academy of Sciences*.

FitzHugh, R. (1961). Impulses and Physiological States in Theoretical Models of Nerve Membrane. *Biophysical Journal*, *1*(6), 445–466.

Fodor, J. A. (1974). Special sciences (or: The disunity of science as a working hypothesis). *Synthese*, *28*(2), 97–115.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, *28*(1), 3-71.

Folia, V., Forkstam, C., Ingvar, M., & Petersson, K. M. (2011). Implicit artificial syntax processing: Genes, preference, and bounded recursion. *Biolinguistics*, *5*(1-2), 105–132.

Frank, S. L. (2006). Strong systematicity in sentence processing by an Echo State Network. In *International Conference on Artificial Neural Networks* (pp. 505–514). Springer.

Frank, S. L., & Fitz, H. (2016). Reservoir computing and the Sooner-is-Better bottleneck. *Behavioral and Brain Sciences*, *39*.

Frank, S. L., Monaghan, P., & Tsoukala, C. (2019). Neural network models of language acquisition and processing. In P. Hagoort (Ed.), *Human Language: From Genes and Brain to Behavior.* Cambridge, USA: MIT Press.

Frank, S. L., Otten, L. J., Galli, G., & Vigliocco, G. (2015). The ERP response to the amount of information conveyed by words in sentences. *Brain and Language*, *140*, 1-11.

Frey, U., & Morris, R. G. (1997). Synaptic tagging and long-term potentiation. *Nature*, *385*(6616), 533–536.

Funahashi, S., Bruce, C. J., & Goldman-Rakic, P. S. (1989). Mnemonic coding of visual space in the monkey's dorsolateral prefrontal cortex. *Journal of Neurophysiology*, *61*(2), 331–349.

Fuster, J. M., & Alexander, G. E. (1971). Neuron activity related to short-term memory. *Science*, *173*(3997), 652–654.

Gallistel, C. R., & King, A. P. (2011). *Memory and the computational brain: Why cognitive science will transform neuroscience* (Vol. 6). John Wiley & Sons.

Ganguli, S., Huh, D., & Sompolinsky, H. (2008). Memory traces in dynamical systems. *Proceedings of the National Academy of Sciences*, *105*(48), 18970–18975.

Gayler, R. W. (2004). Vector Symbolic Architectures answer Jackendoff's challenges for cognitive neuroscience. *arXiv*.

Gerstner, W., Kistler, W. M., Naud, R., & Paninski, L. (2014). *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press.

Gerstner, W., Kreiter, A. K., Markram, H., & Herz, A. V. M. (1997). Neural codes: Firing rates and beyond. *Proceedings of the National Academy of Sciences*, *94*(24), 12740-12741.

Geyer, S., Weiss, M., Reimann, K., Lohmann, G., & Turner, R. (2011). Microstructural Parcellation of the Human Cerebral Cortex – From Brodmann's Post-Mortem Map to in vivo Mapping with High-Field Magnetic Resonance Imaging. *Frontiers in Human Neuroscience*, *5*, 19.

Gibson, E., & Wu, H.-H. I. (2013). Processing Chinese relative clauses in context. *Language and Cognitive Processes*, *28*(1-2), 125-155.

Gibson, J. R., Beierlein, M., & Connors, B. W. (1999). Two networks of electrically coupled inhibitory neurons in neocortex. *Nature*, *402*(6757), 75-79.

Gjorgjieva, J., Drion, G., & Marder, E. (2016). Computational implications of biophysical diversity and multiple timescales in neurons and synapses for circuit performance. *Current Opinion in Neurobiology*, *37*, 44–52.

Goldberg, A. E. (2006). *Constructions at work: The nature of generalization in language*. Oxford University Press.

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., … Agapiou, J. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, *538*(7626), 471.

Günlahar, N., & Kuschner, S. A. (2016). *Using Induced Pluripotent Stem Cells to Model the Pathophysiology of Schizophrenia*. Rotterdam: Erasmus University Rotterdam.

Hadley, R. F. (1994). Systematicity in Connectionist Language Learning. *Mind & Language*, *9*(3), 247-272.

Hadley, R. F. (2006). Neural circuits, matrices, and conjunctive binding. *Behavioral and Brain Sciences*, *29*(1), 80-80.

Hadley, R. F., Rotaru-Varga, A., Arnold, D. V., & Cardei, V. C. (2001). Syntactic systematicity arising from semantic predictions in a Hebbian-competitive network. *Connection Science*, *13*(1), 73–94.

Haeusler, S., & Maass, W. (2007). A Statistical Analysis of Information-Processing Properties of Lamina-Specific Cortical Microcircuit Models. *Cerebral Cortex*, *17*(1), 149-162.

Hagoort, P. (2003). How the brain solves the binding problem for language: A neurocomputational model of syntactic processing. *NeuroImage*, *20*, S18-S29.

Hagoort, P. (2005). On Broca, brain, and binding: A new framework. *Trends in Cognitive Sciences*, *9*(9), 416-423.

Hagoort, P. (2013). MUC (memory, unification, control) and beyond. *Frontiers in Psychology*, *4*, 416.

Hagoort, P., Hald, L., Bastiaansen, M., & Petersson, K. M. (2004). Integration of Word Meaning and World Knowledge in Language Comprehension. *Science*, *304*(5669), 438-441.

Hasson, U., Chen, J., & Honey, C. J. (2015). Hierarchical process memory: Memory as an integral component of information processing. *Trends in Cognitive Sciences*, *19*(6), 304-313.

Hickok, G., & Poeppel, D. (2007). The cortical organization of speech processing. *Nature Reviews Neuroscience*, *8*(5), 393-402.

Hodgkin, A. L., & Huxley, A. F. (1952). Currents carried by sodium and potassium ions through the membrane of the giant axon of Loligo. *The Journal of Physiology*, *116*(4), 449-472.

Huettig, F. (2015). Four central questions about prediction in language processing. *Brain Research*, *1626*, 118-135.

Hummel, J. E. (2011). Getting symbols out of a neural architecture. *Connection Science*, *23*(2), 109-118.

Ippolito, D. M., & Eroglu, C. (2010). Quantifying Synapses: An Immunocytochemistry-based Assay to Quantify Synapse Number. *Journal of Visualized Experiments : JoVE*(45), e2270.

Isbister, J. B., Eguchi, A., Ahmad, N., Galeazzi, J. M., Buckley, M. J., & Stringer, S. (2018). A new approach to solving the feature-binding problem in primate vision. *Interface Focus*, *8*(4), 20180021.

Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, *14*(6), 1569–1572.

Izhikevich, E. M. (2006). Polychronization: computation with spikes. *Neural Computation*, *18*(2), 245–282.

Izhikevich, E. M., & Edelman, G. M. (2008). Large-scale model of mammalian thalamocortical systems. *Proceedings of the National Academy of Sciences*, *105*(9), 3593-3598.

Jackendoff, R. (1992). *Semantic structures* (Vol. 18). MIT press.

Jackendoff, R. (2002). *Foundations of Language: Brain, Meaning, Grammar, Evolution*. Oxford University Press.

Jackendoff, R. (2007). A Parallel Architecture perspective on language processing. *Brain Research*, *1146*, 2-22.

Jacob, B., Ng, S., & Wang, D. (2010). *Memory systems: Cache, DRAM, disk*. Morgan Kaufmann.

Jaeger, H. (2001). The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, *148*(34), 13.

Joanisse, M. F., & Seidenberg, M. S. (1999). Impairments in verb morphology after brain injury: A connectionist model. *Proceedings of the National Academy of Sciences*, *96*(13), 7592-7597.

Jolivet, R., Kobayashi, R., Rauch, A., Naud, R., Shinomoto, S., & Gerstner, W. (2008). A benchmark test for a quantitative assessment of simple neuron models. *Journal of Neuroscience Methods*, *169*(2), 417-424.

Kempter, R., Gerstner, W., & van Hemmen, J. L. (1999). Hebbian learning and spiking neurons. *Physical Review E*, *59*(4), 4498-4514.

Koch, C. (1999). *Biophysics of computation: Information processing in single neurons*. Oxford University Press.

Kubota, K., & Niki, H. (1971). Prefrontal cortical unit activity and delayed alternation performance in monkeys. *Journal of Neurophysiology*, *34*(3), 337–347.

Kumar, A., Rotter, S., & Aertsen, A. (2010). Spiking activity propagation in neuronal networks: Reconciling different perspectives on neural coding. *Nature Reviews Neuroscience*, *11*(9), 615-627.

Kuperberg, G. R., & Jaeger, T. F. (2016). What do we mean by prediction in language comprehension? *Language, Cognition and Neuroscience*, *31*(1), 32–59.

Kuperberg, G. R., Paczynski, M., & Ditman, T. (2011). Establishing causal coherence across sentences: An ERP study. *Journal of Cognitive Neuroscience*, *23*(5), 1230–1246.

Kutas, M., & Federmeier, K. D. (2011). Thirty Years and Counting: Finding Meaning in the N400 Component of the Event-Related Brain Potential (ERP). *Annual Review of Psychology*, *62*(1), 621-647.

Kutas, M., & Hillyard, S. A. (1980). Reading senseless sentences: Brain potentials reflect semantic incongruity. *Science*, *207*(4427), 203–205.

Legenstein, R., Naeger, C., & Maass, W. (2005). What Can a Neuron Learn with Spike-Timing-Dependent Plasticity? *Neural Computation*, *17*(11), 2337-2382.

Lewis, R. L., Vasishth, S., & Van Dyke, J. A. (2006). Computational principles of working memory in sentence comprehension. *Trends in Cognitive Sciences*, *10*(10), 447–454.

Li, K. W., Klemmer, P., & Smit, A. B. (2010). Interaction proteomics of synapse protein complexes. *Analytical and Bioanalytical Chemistry*, *397*(8), 3195-3202.

Litwin-Kumar, A., & Doiron, B. (2014). Formation and maintenance of neuronal assemblies through synaptic plasticity. *Nature Communications*, *5*, 5319.

London, M., Roth, A., Beeren, L., Häusser, M., & Latham, P. (2010). Sensitivity to perturbations in vivo implies high noise and suggests rate coding in cortex. *Nature*, *466*(7302), 123-127.

Lukoševičius, M., & Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, *3*(3), 127–149.

Lukoševičius, M., Jaeger, H., & Schrauwen, B. (2012). Reservoir Computing Trends. *KI - Künstliche Intelligenz*, *26*(4), 365-371.

Lundqvist, M., Rose, J., Herman, P., Brincat, S. L., Buschman, T. J., & Miller, E. K. (2016). Gamma and beta bursts underlie working memory. *Neuron*, *90*(1), 152–164.

Luo, L. (2015). *Principles of Neurobiology*. Garland Science.

Maass, W., Legenstein, R. A., & Bertschinger, N. (2005). Methods for estimating the computational power and generalization capability of neural microcircuits. In *Advances in Neural Information Processing Systems* (pp. 865–872).

Maass, W., Natschläger, T., & Markram, H. (2002). Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. *Neural Computation*, *14*(11), 2531-2560.

Markram, H., Lübke, J., Frotscher, M., & Sakmann, B. (1997). Regulation of Synaptic Efficacy by Coincidence of Postsynaptic APs and EPSPs. *Science*, *275*(5297), 213-215.

Markram, H., Muller, E., Ramaswamy, S., Reimann, M. W., Abdellah, M., Sanchez, C. A., … Schürmann, F. (2015). Reconstruction and Simulation of Neocortical Microcircuitry. *Cell*, *163*(2), 456-492.

Markram, H., Toledo-Rodriguez, M., Wang, Y., Gupta, A., Silberberg, G., & Wu, C. (2004). Interneurons of the neocortical inhibitory system. *Nature Reviews Neuroscience*, *5*(10), 793-807.

Markram, H., Wang, Y., & Tsodyks, M. (1998). Differential signaling via the same axon of neocortical pyramidal neurons. *Proceedings of the National Academy of Sciences*, *95*(9), 5323–5328.

Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information. MIT Press. *Cambridge, Massachusetts*.

Marslen-Wilson, W. (1975). Sentence Perception as an Interactive Parallel Process. *Science*, *189*(4198), 226-228.

Marslen-Wilson, W., & Tyler, L. K. (1980). The temporal structure of spoken language understanding. *Cognition*, *8*(1), 1-71.

Martin, A. E., & Doumas, L. A. A. (2017). A mechanism for the cortical computation of hierarchical linguistic structure. *PLOS Biology*, *15*(3), e2000663.

Martin, K. C., & Kosik, K. S. (2002). Synaptic tagging — who's it? *Nature Reviews Neuroscience*, *3*(10), 813-820.

Masquelier, T., Guyonneau, R., & Thorpe, S. J. (2008). Competitive STDP-Based Spike Pattern Learning. *Neural Computation*, *21*(5), 1259-1276.

Mazzoni, A., Lindén, H., Cuntz, H., Lansner, A., Panzeri, S., & Einevoll, G. T. (2015). Computing the Local Field Potential (LFP) from Integrate-and-Fire Network Models. *PLOS Computational Biology*, *11*(12), e1004584.

McClelland, J. L., & Elman, J. L. (1986). The TRACE model of speech perception.

Cognitive Psychology, *18*(1), 1-86.

Memmesheimer, R.-M., Rubin, R., Ölveczky, B. P., & Sompolinsky, H. (2014). Learning Precisely Timed Spikes. *Neuron*, *82*(4), 925-938.

Meyer, A., Wheeldon, L., & Krott, A. (2007). *Automaticity and Control in Language Processing*. Psychology Press.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, *63*(2), 81.

Mongillo, G., Barak, O., & Tsodyks, M. (2008). Synaptic theory of working memory. *Science*, *319*(5869), 1543–1546.

Morris, R. K. (1994). Lexical and message-level sentence context effects on fixation times in reading. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *20*(1), 92.

Morrison, A., Diesmann, M., & Gerstner, W. (2008). Phenomenological models of synaptic plasticity based on spike timing. *Biological Cybernetics*, *98*(6), 459-478.

Murray, J. D., Bernacchia, A., Freedman, D. J., Romo, R., Wallis, J. D., Cai, X., … Wang, X.-J. (2014). A hierarchy of intrinsic timescales across primate cortex. *Nature Neuroscience*, *17*, 1661–1663.

Naud, R., Marcille, N., Clopath, C., & Gerstner, W. (2008). Firing patterns in the adaptive exponential integrate-and-fire model. *Biological Cybernetics*, *99*(4-5), 335.

Nieuwland, M. S., Politzer-Ahles, S., Heyselaar, E., Segaert, K., Darley, E., Kazanina, N., … Ito, A. (2018). Large-scale replication study reveals a limit on probabilistic prediction in language comprehension. *eLife*, *7*, e33468.

Nunez, P. L., & Srinivasan, R. (2006). *Electric fields of the brain: The neurophysics of EEG*. Oxford University Press.

Okun, M., Steinmetz, N. A., Cossell, L., Iacaruso, M. F., Ko, H., Barthó, P., … Harris, K. D. (2015). Diverse coupling of neurons to populations in sensory cortex. *Nature*, *521*(7553), 511-515.

Ostojic, S. (2014). Two types of asynchronous activity in networks of excitatory and inhibitory spiking neurons. *Nature Neuroscience*, *17*(4), 594-600.

Perin, R., Berger, T. K., & Markram, H. (2011). A synaptic organizing principle for cortical neuronal groups. *Proceedings of the National Academy of Sciences*, *108*(13), 5419-5424.

Petersson, K. M. (2005). On the relevance of the neurobiological analogue of the finite-state architecture. *Neurocomputing*, *65*, 825–832.

Petersson, K.-M., Folia, V., & Hagoort, P. (2012). What artificial grammar learning reveals about the neurobiology of syntax. *Brain and Language*, *120*(2), 83–95.

Petersson, K. M., & Hagoort, P. (2012). The neurobiology of syntax: Beyond string sets. *Phil. Trans. R. Soc. B*, *367*(1598), 1971–1983.

Petersson, K. M., Simos, T. E., & Psihoyios, G. (2008). On Cognition, Structured Sequence Processing, and Adaptive Dynamical Systems. In *AIP Conference Proceedings* (p. 195-199). AIP.

Pfister, J.-P., & Gerstner, W. (2006). Triplets of Spikes in a Model of Spike Timing-Dependent Plasticity. *Journal of Neuroscience*, *26*(38), 9673-9682.

Plate, T. A. (1995). Holographic reduced representations. *IEEE Transactions on Neural Networks*, *6*(3), 623-641.

Plunkett, K., & Marchman, V. (1991). U-shaped learning and frequency effects in a multi-layered perception: Implications for child language acquisition. *Cognition*, *38*(1), 43-102.

Pulvermüller, F. (2010). Brain embodiment of syntax and grammar: Discrete combinatorial mechanisms spelt out in neuronal circuits. *Brain and Language*, *112*(3), 167-179.

Rabagliati, H., Doumas, L. A. A., & Bemis, D. K. (2017). Representing composed meanings through temporal binding. *Cognition*, *162*, 61-72.

Rabinovich, M., Volkovskii, A., Lecanda, P., Huerta, R., Abarbanel, H. D. I., & Laurent, G. (2001). Dynamical encoding by networks of competing neuron groups: Winnerless competition. *Physical Review Letters*, *87*(6), 068102.

Rauch, A., Camera, G. L., Lüscher, H.-R., Senn, W., & Fusi, S. (2003). Neocortical pyramidal cells respond as integrate-and-fire neurons to in vivo–like input currents. *Journal of Neurophysiology*, *90*(3), 1598-1612.

Ravichandran, D., & Hovy, E. (2002). Learning surface text patterns for a question answering system. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 41–47). Stroudsburg, PA, USA: Association for Computational Linguistics.

Reimann, M. W., King, J. G., Muller, E. B., Ramaswamy, S., & Markram, H. (2015). An algorithm to predict the connectome of neural microcircuits. *Frontiers in Computational Neuroscience*, *9*, 28.

Rieke, F., & Warland, D. (1999). *Spikes: Exploring the neural code*. Cambridge, USA: MIT press.

Rochester, N., Holland, J., Haibt, L., & Duda, W. (1956). Tests on a cell assembly theory of the action of the brain, using a large digital computer. *IRE Transactions on Information Theory*, *2*(3), 80-93.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533.

Seidenberg, M. S. (1993). Connectionist Models and Cognitive Theory. *Psychological Science*, *4*(4), 228-235.

Shadlen, M. N., & Newsome, W. T. (1998). The variable discharge of cortical neurons: Implications for connectivity, computation, and information coding. *Journal of Neuroscience*, *18*(10), 3870–3896.

Shastri, L., & Ajjanagadde, V. (1993). From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences*, *16*(3), 417–451.

Siegelmann, H. T. (1999). *Neural networks and analog computation: Beyond the turing limit*. Birkhäuser.

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, *46*(1-2), 159–216.

Softky, W. R., & Koch, C. (1993). The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs. *Journal of Neuroscience*, *13*(1), 334–350.

Song, S., Sjöström, P. J., Reigl, M., Nelson, S., & Chklovskii, D. B. (2005). Highly Nonrandom Features of Synaptic Connectivity in Local Cortical Circuits. *PLOS Biology*, *3*(3), e68.

Sougné, J. P. (1998). Connectionism and the problem of multiple instantiation. *Trends in Cognitive Sciences*, *2*(5), 183-189.

Sougné, J. P. (2001). Binding and multiple instantiation in a distributed network of spiking nodes. *Connection Science*, *13*(2), 99–126.

Sougné, J. P. (2006). Binding Problem. In *Encyclopedia of Cognitive Science.* American Cancer Society.

Sporns, O., & Kötter, R. (2004). Motifs in Brain Networks. *PLOS Biology*, *2*(11), e369.

Sporns, O., Tononi, G., & Kötter, R. (2005). The Human Connectome: A Structural Description of the Human Brain. *PLOS Computational Biology*, *1*(4), e42.

Sreenivasan, K. K., & D'Esposito, M. (2019). The what, where and how of delay activity. *Nature Reviews Neuroscience*, *20*(8), 466–481.

St. John, M. F., & McClelland, J. L. (1990). Learning and applying contextual constraints in sentence comprehension. *Artificial Intelligence*, *46*(1), 217–257.

Stokes, M. G. (2015). 'Activity-silent'working memory in prefrontal cortex: A dynamic coding framework. *Trends in Cognitive Sciences*, *19*(7), 394–405.

Strata, P., & Harvey, R. (1999). Dale's principle. *Brain Research Bulletin*, *50*(5), 349-350.

Strotzer, M. (2009). One Century of Brain Mapping Using Brodmann Areas*. *Clinical Neuroradiology*, *19*(3), 179-186.

Sturt, P., & Lombardo, V. (2005). Processing Coordinated Structures: Incrementality and Connectedness. *Cognitive Science*, *29*(2), 291-305.

Takeuchi, T., Duszkiewicz, A. J., & Morris, R. G. (2014). The synaptic plasticity and memory hypothesis: Encoding, storage and persistence. *Phil. Trans. R. Soc. B*, *369*(1633), 20130288.

Thorpe, S., Delorme, A., & Van Rullen, R. (2001). Spike-based strategies for rapid processing. *Neural Networks*, *14*(6-7), 715-725.

Titley, H. K., Brunel, N., & Hansel, C. (2017). Toward a neurocentric view of learning. *Neuron*, *95*(1), 19–32.

Traxler, M. J., & Foss, D. J. (2000). Effects of sentence constraint on priming in natural language comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *26*(5), 1266.

Tsodyks, M. V., & Sejnowski, T. (1995). Rapid state switching in balanced cortical network models. *Network: Computation in Neural Systems*, *6*(2), 111–124.

Tuckwell, H. C. (1988). *Introduction to theoretical neurobiology. Vol. 2, volume 8 of Cambridge Studies in Mathematical Biology*. Cambridge University Press.

Turrigiano, G. G., Leslie, K. R., Desai, N. S., Rutherford, L. C., & Nelson, S. B. (1998). Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature*, *391*(6670), 892-896.

van den Broek, D., Uhlmann, M., Duarte, R., Fitz, H., Hagoort, P., & Petersson, K. M. (2018). The Best Spike Filter Kernel is a Neuron. In *Extended Abstracts to the Cognitive Computational Neuroscience Conference.* New York, NY.

van der Velde, F., & de Kamps, M. (2015). The necessity of connection structures in neural models of variable binding. *Cognitive Neurodynamics*, *9*(4), 359-370.

van der Velde, F., & de Kamps, M. (2006). Neural blackboard architectures of combinatorial structures in cognition. *Behavioral and Brain Sciences*, *29*(1), 37-70.

van Essen, D. C., Drury, H. A., Joshi, S., & Miller, M. I. (1998). Functional and structural mapping of human cerebral cortex: Solutions are in the surfaces. *Proceedings of the National Academy of Sciences*, *95*(3), 788–795.

van Gompel, R. P. G., Pickering, M. J., Pearson, J., & Liversedge, S. P. (2005). Evidence against competition during syntactic ambiguity resolution. *Journal of Memory and Language*, *52*(2), 284-307.

van Petten, C. (1993). A comparison of lexical and sentence-level context effects in event-related potentials. *Language and Cognitive Processes*, *8*(4), 485-531.

van Petten, C., & Luka, B. J. (2012). Prediction during language comprehension: Benefits, costs, and ERP components. *International Journal of Psychophysiology*, *83*(2), 176-190.

van Vreeswijk, C., & Sompolinsky, H. (1996). Chaos in neuronal networks with balanced excitatory and inhibitory activity. *Science*, *274*(5293), 1724–1726.

van den Heuvel, M. P., & Sporns, O. (2011). Rich-Club Organization of the Human Connectome. *Journal of Neuroscience*, *31*(44), 15775-15786.

Varga, A. P., & Moore, R. K. (1990). Hidden Markov model decomposition of speech and noise. In *International Conference on Acoustics, Speech, and Signal Processing* (p. 845-848 vol.2).

Vosse, T., & Kempen, G. (2000). Syntactic structure assembly in human parsing: A computational model based on competitive inhibition and a lexicalist grammar. *Cognition*, *75*(2), 105–143.

Wallace, E., Maei, H. R., & Latham, P. E. (2013). Randomly connected networks have short temporal memory. *Neural Computation*, *25*(6), 1408–1439.

Wang, X.-J. (2002). Probabilistic Decision Making by Slow Reverberation in Cortical Circuits. *Neuron*, *36*(5), 955-968.

Wang, X.-J., Tegnér, J., Constantinidis, C., & Goldman-Rakic, P. S. (2004). Division of labor among distinct subtypes of inhibitory neurons in a cortical microcircuit of working memory. *Proceedings of the National Academy of Sciences*, *101*(5), 1368-1373.

Watanabe, K., & Funahashi, S. (2014). Neural mechanisms of dual-task interference and cognitive capacity limitation in the prefrontal cortex. *Nature Neuroscience*, *17*(4), 601.

Whittington, J. C. R., & Bogacz, R. (2019). Theories of error back-propagation in the brain. *Trends in Cognitive Sciences*, *23*(3), 235–250.

Zenke, F., Agnes, E. J., & Gerstner, W. (2015). Diverse synaptic plasticity mechanisms orchestrated to form and retrieve memories in spiking neural networks. *Nature Communications*, *6*(1), 1–13.

Appendix

A

English summary

Nederlandse samenvatting

Deutsche Zusammenfassung

Acknowledgements

Biography

MPI Series in Psycholinguistics

A.1 English summary

Studying the human brain and mind poses many unique challenges. One of them is that different scientific rules and laws are suitable to describe the functioning of different aspects of the brain on different levels of granularity. This is often not the case in other sciences, for example, the apple falling from the tree is governed by the same law as the earth revolving around the sun. The physics of water whirling down a drain is not all that different from the way a hurricane works. In the brain, however, the way two brain cells, two neurons, interact has very little to do with the description of, for example, how humans form a sentence. And that is the case even though we know for a fact that the interaction between neurons, when taking millions and billions of them, is the exact thing that gives rise to us forming a sentence.

Over the last decades, the different levels of granularity of the brain have been studied individually in detail. For example, there is a rich body of work describing and investigating the properties of language or how syntax and semantics in a sentence work. Furthermore, our understanding of the role that different brain areas play has progressed significantly – today, we know that each part of the brain has its own distinct role to play in our processing and, for example, that there is one brain area dedicated to processing the syntax of sentences. Similarly, the study of the smallest components of the brain, the brain's neurons, has made great progress in recent decades. Through measuring, analyzing, and modeling neurons individually, we now have a good understanding of how neurons work, how they communicate, and how they respond to input. They are considered the underlying fabric of how the human brain computes, uniting in themselves aspects of a computer's CPU and memory at the same time.

However, there are still significant gaps in our understanding of how these different levels of granularity interact. For example, how do our neurons store and maintain the meaning and properties of words? How do they maintain information over the time of a sentence and a conversation? How are sentences analyzed, disassembled, and interpreted given the variety of different meanings that similar sentences can have and the variety of similar meanings that very different sentences can have?

In this doctoral thesis, I aim to contribute to filling in the gap of knowledge between the smallest level, the neuron, and the highest level, in particular language. For this, I study the behavior of the smallest biological unit, neurons, and how they interact with each other in small and intermediate networks, while maintaining a conceptual connection to the level of the human brain as a whole, thinking about how humans use and understand language. To do this, I applied a method that we call *neurobiological modeling*. I simulated important properties of individual neurons, which have been identified by neurobiology, in a computer simulation. I let neurons interact in a small network of a few thousand exemplars and then I train them to solve tasks that are relevant for linguistics. In the different chapters of this dissertation, I used different tasks since they have different targets of study. For example, to study memory properties, I fed a word into the network and tested after how much time the information of this word was still retained. To study sentence processing, I input a sentence like "The red cat chases the brown dog." and analyzed if it was possible for the network to identify who was chasing, who was being chased, and what color each of them was. I found that the neural network I was simulating is well suited to solve these tasks. Since I used a computer simulation, instead of biological neurons, I could freely manipulate it and analyze how exactly the task was solved. Through this, I could study the mechanisms within neurons that give rise to a particular linguistic computation. For example, I found that success in a short term memory task, i.e. how the network remembers a word from the beginning of a sentence at the end of a sentence, crucially depends on a neuronal property that is called *spike rate adaptation*. When this was absent, the task was impossible to solve. Since the simulation was done in a neurobiological model, meaning in a manner that is similar to the neurons in our brain, one can conclude that spike rate adaptation must also be important for our brain to maintain information during sentence processing. Through a similar chain of arguments, I could show that processing sentences, dissecting their meaning, and deciding if two sentences have a similar or different meaning, irrespective of their syntactic structure, depends on a rich set of dynamical properties in each neuron. These include spike rate adaptation but also, for example, how precisely the neurons communicate with each other (i.e. the strength and shape of synaptic currents).

With the neurobiological models, my dissertation provides a specialized tool to investigate our brain's computation on the level of neurons performing processing tasks that are directly relevant for language processing. Ultimately, our brains are exquisitely complex systems and processing not just of language depends on the interactions between neurons but also the properties of the neurons itself, parts of the neurons, and even individual molecules within them. Each element plays an important role for our brains to successfully process information. And the only way to understand our brain is to not just understand individual elements but how they all work in unison.

A.2 Nederlandse samenvatting

Het bestuderen van het menselijk brein en de geest brengt veel unieke uitdagingen met zich mee. Een daarvan is dat op verschillende niveaus van granulariteit, verschillende wetenschappelijke regels en wetten geschikt zijn om het te beschrijven. Dit is in andere wetenschappen vaak niet het geval, de appel die van de boom valt, valt bijvoorbeeld onder dezelfde wet als de aarde die om de zon draait. De fysica van water dat door een afvoer wervelt, verschilt niet zo heel veel van de manier waarop een orkaan werkt. In de hersenen heeft de manier waarop twee hersencellen, twee neuronen, op elkaar inwerken echter weinig te maken met de beschrijving van bijvoorbeeld hoe mensen een zin vormen. En dat is het geval, ook al weten we zeker dat de interactie van die twee kleine neuronen, wanneer men er miljoenen en miljarden van neemt, precies datgene is dat aanleiding geeft tot het vormen van een zin.

In de afgelopen decennia zijn de verschillende niveaus van granulariteit van de hersenen op individuele basis in detail bestudeerd. Er is bijvoorbeeld een rijke hoeveelheid werk dat de eigenschappen van talen beschrijft en onderzoekt, alsmede hoe syntaxis en semantiek in een zin werken. Bovendien is ons begrip van de rol die verschillende delen van onze hersenen spelen aanzienlijk verbeterd - vandaag weten we dat elk deel van de hersenen zijn eigen specifieke rol te spelen heeft in onze cognitie en, bijvoorbeeld, dat er één hersengebied is dat gespecialiseerd is in de verwerking van de syntaxis van zinnen. Tegelijkertijd heeft de studie van de kleinste componenten van de hersenen, de neuronen, de afgelopen decennia grote vooruitgang geboekt. Door neuronen afzonderlijk te meten, te analyseren en te modelleren, hebben we nu een goed begrip van hoe neuronen werken, hoe ze communiceren en hoe ze reageren op input. Ze worden beschouwd als het onderliggende fundament van hoe het menselijk brein informatie verwerkt, waarbij aspecten van het CPU en het geheugen van een computer tegelijkertijd worden verenigd.

Er zijn echter nog steeds aanzienlijke hiaten in ons begrip van hoe deze verschillende niveaus van granulariteit op elkaar inwerken. Hoe slaan onze neuronen bijvoorbeeld de betekenis en eigenschappen van woorden op? Hoe behouden ze deze informatie voor de duur van een zin of een gesprek? Hoe worden zinnen geanalyseerd, uit elkaar gehaald en geïnterpreteerd, gezien het feit dat vergelijkbare zinnen verschillende betekenissen kunnen hebben en dat zeer verschillende zinnen vergelijkbare betekenissen kunnen hebben?

In dit proefschrift wil ik bijdragen aan het dichten van de kenniskloof tussen het kleinste niveau, het neuron en het hoogste niveau, in het bijzonder taal. Hiervoor bestudeer ik het gedrag van de kleinste biologische eenheid, het neuron, en hoe ze met elkaar omgaan in netwerken met een kleine tot gemiddelde grootte, terwijl ik een conceptuele verbinding met het niveau van het menselijk brein als geheel behoud, denkend aan hoe mensen taal gebruiken en begrijpen. Om dit te doen, heb ik een methode toegepast die we *neurobiologisch modelleren* noemen. Ik simuleerde belangrijke eigenschappen van individuele neuronen, die door neurobiologie zijn geïdentificeerd, in een computersimulatie. Ik laat neuronen samenwerken in een klein netwerk van een paar duizend exemplaren en daarna train ik ze om taken op te lossen die relevant zijn voor de taalkunde. In de verschillende hoofdstukken van dit proefschrift heb ik verschillende taken gebruikt, elk met een ander studiedoel. Om bijvoorbeeld de eigenschappen van het geheugen te bestuderen, voer ik een woord in het netwerk in en test ik na hoeveel tijd de informatie van dit woord nog wordt bewaard. Om de verwerking van zinnen te bestuderen, voer ik een zin in als "De rode kat jaagt op de bruine hond." en analyseren of het mogelijk is voor het netwerk om te bepalen wie er achterna zit, wie er wordt achtervolgd en welke kleur elk van hen heeft. Ik ontdekte dat het neurale netwerk dat ik simuleerde, zeer geschikt is om deze taken op te lossen. Omdat ik een computersimulatie gebruikte, in plaats van biologische neuronen, kon ik ze vrijelijk manipuleren en kon ik analyseren hoe de taak precies was opgelost. Hierdoor kon ik het mechanisme in het neuron bestuderen dat aanleiding geeft tot een bepaalde taalkundige computatie. Ik ontdekte bijvoorbeeld dat succes bij een korte-termijn geheugentaak, d.w.z. hoe het een woord uit het begin van een zin onthoudt aan het einde van een zin, cruciaal afhangt van een neuronale eigenschap die *spike rate adaptation* wordt genoemd. Als dit afwezig was, was de taak niet op te lossen. Aangezien de simulatie werd uitgevoerd in een neurobiologisch model, dat wil zeggen op een manier die vergelijkbaar is met de neuronen in onze hersenen, kan worden geconcludeerd dat aanpassing van de spike rate adaptation ook belangrijk moet zijn voor onze hersenen om informatie te behouden tijdens het verwerken van zinnen. Door een soortgelijke reeks argumenten kon ik aantonen dat het verwerken van zinnen afhangt van een rijke reeks dynamische eigenschappen in elk neuron. Dit omvat aanpassing van de piekfrequentie, maar bijvoorbeeld ook hoe neuronen precies met elkaar communiceren (d.w.z. de sterkte en vorm van synaptische stroom).

Mijn proefschrift biedt met de neurobiologische modellen een gespecialiseerd hulpmiddel om de berekeningen van onze hersenen te onderzoeken op het niveau van neuronen die verwerkingstaken uitvoeren die direct relevant zijn voor taalverwerking. Uiteindelijk zijn onze hersenen buitengewoon complexe machines en de verwerking van niet alleen taal hangt af van de interacties tussen neuronen maar ook van de eigenschappen van de neuronen zelf, delen van de neuronen en zelfs individuele moleculen erin. Elk element speelt een belangrijke rol voor onze hersenen om informatie succesvol te verwerken. En de enige manier om ons brein te begrijpen, is door niet alleen individuele elementen te begrijpen, maar ook hoe ze allemaal samenwerken.

Translated by Rowan Sommers

A.3 Deutsche Zusammenfassung

Beim Studium des menschlichen Gehirns stellen sich viele einzigartige Herausforderungen. Eine davon ist, dass für verschiedene Aspekte und für eine unterschiedliche Detailtiefe der Beschreibung, unterschiedliche wissenschaftliche Regeln und Modelle notwendig sind. Dies ist in anderen Wissenschaften oft nicht der Fall, zum Beispiel unterliegt der Apfel der vom Baum fällt demselben Gesetz wie die Erde, die sich um die Sonne dreht. Die Physik des Wassers, das einen Abfluss hinunterwirbelt, unterscheidet sich nicht wesentlich von der Funktionsweise eines Hurrikans. Im Gegensatz dazu hat die Art und Weise, wie im Gehirn zwei Gehirnzellen interagieren, relativ wenig mit unserem Verständnis davon zu tun, wie beispielsweise Menschen einen Satz bilden. Das ist jedoch der Fall, obwohl wir sicher wissen, dass die Interaktion einzelner Neuronen, wenn man Millionen und Milliarden von ihnen nimmt, genau das ist, was dazu führt, dass wir überhaupt einen Satz bilden können.

In den letzten Jahrzehnten wurden verschiedene Aspekte des Gehirns unabhängig voneinander erforscht. Zum Beispiel wurde sehr genau untersucht, wie Sprache funktioniert und wie z. B. Syntax und Semantik in einem Satz interagieren. Darüber hinaus haben wir heute ein gutes Verständnis davon, welche Rolle verschiedene Areale unseres Gehirns spielen. Wir wissen, dass jeder Teil des Gehirns eine eigene Rolle bei der Verarbeitung von Informationen spielt und dass es beispielsweise ein Areal gibt, das der Verarbeitung der Syntax in einem Satz gewidmet ist. Gleichzeitig hat die Untersuchung der kleinsten Bestandteile des Gehirns, der Neuronen, in den letzten Jahrzehnten große Fortschritte gemacht. Durch das individuelle Messen, Analysieren und Modellieren von Neuronen haben wir heute ein gutes Verständnis davon, wie Neuronen funktionieren, wie sie kommunizieren und wie sie auf Input reagieren. Sie gelten als die Grundlage für die Informationsverarbeitung des menschlichen Gehirns und vereinen Eigenschaften sowohl von Computerprozessoren als auch von Arbeitsspeichern.

Es gibt jedoch immer noch erhebliche Lücken in unserem Verständnis, wie diese verschiedenen Aspekte interagieren. Wie speichern unsere Neuronen beispielsweise die Bedeutung und Eigenschaften von Wörtern? Wie behalten sie Informationen über die Dauer eines Satzes und eines Gesprächs? Wie werden Sätze analysiert, zerlegt und interpretiert? Und wie können ähnliche Sätze, eine Vielfalt unterschiedlicher Bedeutungen haben können, und sehr unterschiedliche Sätze, die viele ähnlichen Bedeutungen haben können, korrekt interpretiert?

Mit dieser Doktorarbeit möchte ich dazu beitragen, die Wissenslücke zwischen detaillierten und abstrakteren Betrachtungsweisen, insbesondere von Neuronen und Sprache, zu schließen. Dazu untersuche ich das Verhalten der kleinsten biologischen Einheiten, der Neuronen, und wie sie in kleinen und mittleren Netzwerken miteinander interagieren. Gleichzeitig halte ich eine konzeptionelle Verbindung zur Ebene des menschlichen Gehirns als Ganzes aufrecht und betrachte wie Menschen Sprache verwenden und verstehen. Dazu habe ich eine Methode angewandt, die wir *neurobiologische Modellierung* nennen. In einer Computersimulation habe ich wichtige Eigenschaften einzelner Neuronen simuliert, die in der Neurobiologie identifiziert wurden. Ich lasse Neuronen in einem kleinen Netzwerk von einigen tausend Exemplaren interagieren und trainiere sie dann darauf, bestimmte Aufgaben zu lösen, die für die Linguistik relevant sind. In den einzelnen Kapiteln dieser Dissertation habe ich verschiedene Aufgaben verwendet, da es unterschiedliche Untersuchungsziele gibt. Um beispielsweise die Gedächtniseigenschaften zu untersuchen, gebe ich ein Wort in dieses Netzwerk ein und teste, wie lange die Informationen dieses Wortes noch verfügbar sind. Um die Satzverarbeitung zu studieren, sende ich einen Satz wie "Die rote Katze jagt den braunen Hund.", um zu analysieren, ob es dem Netzwerk möglich ist zu identifizieren, wer jagt, beziehungsweise wer gejagt wird und welche Farbe jeder von ihnen hat. Ich fand heraus, dass das neuronale Netzwerk, das ich simulierte, gut geeignet ist, diese Aufgaben zu lösen. Da ich anstelle von biologischen Neuronen eine Computersimulation verwendet habe, konnte ich sie flexibel manipulieren und analysieren, wie die Aufgabe im Detail gelöst wurde. Dadurch konnte ich den Mechanismus innerhalb der Neuronen untersuchen, der zu einer bestimmten sprachlichen Verarbeitung führt. Ich fand ich heraus, dass der Erfolg einer Kurzzeitgedächtnisaufgabe, d. h. wie das Netzwerk sich an ein Wort vom Anfang eines Satzes am Ende dieses Satzes erinnert, entscheidend von einer neuronalen Eigenschaft abhängt, die als *Feuerratenadaption* bezeichnet wird. Wenn diese nicht vorhanden war, konnte die Aufgabe nicht gelöst werden. Die Simulation wurde in einem neurobiologischen Modell durchgeführt, also in einer Weise, die der Funktionsweise der Neuronen in unserem Gehirn ähnlich ist. Daher kann man daraus schließen, dass die Feuerratenadaption auch wichtig sein muss, damit unser Gehirn Informationen während der Satzverarbeitung behält. Durch eine ähnliche Argumentationskette konnte ich zeigen, dass die Verarbeitung von Sätzen, die Analyse ihrer Bedeutung und die Entscheidung, ob zwei Sätze unabhängig von ihrer syntaktischen Struktur eine ähnliche oder unterschiedliche Bedeutung haben, von einer Vielzahl dynamischer Eigenschaften in jedem Neuron abhängt. Dies umfasst die Feuerratenadaption, aber auch beispielsweise, wie genau Neuronen miteinander kommunizieren (d. h. die Stärke und Form von synaptischen Strömen).

Meine Dissertation bietet mit den neurobiologischen Modellen ein passendes Werkzeug, um zu untersuchen, wie unser Gehirn auf der Ebene von Neuronen Informationen in einer Art verarbeitet, die direkt für Sprachverarbeitung relevant ist. Letztendlich ist unser Gehirn ein äußerst komplexes System und die Verarbeitung nicht nur der Sprache hängt von den Interaktionen zwischen Neuronen ab, aber auch von den Eigenschaften der Neuronen selbst, Teilen der Neuronen und sogar einzelner Moleküle in ihnen. Jede Einheit spielt eine wichtige Rolle für unser Gehirn, um Informationen erfolgreich zu verarbeiten. Der einzige Weg, unser Gehirn zu verstehen, besteht darin, nicht nur einzelne Elemente zu verstehen, sondern auch wie sie alle zusammenarbeiten.

A.4 Acknowledgements

First of all, I thank my supervisors, Karl Magnus and Hartmut, without whom this thesis obviously would not have been possible. When I first read the PhD call, one of the things that drew me most to this project is that it takes mathematical analysis seriously while keeping the conceptual analysis center stage. That it dares to stay mathematically precise where the discrepancy between explanatory levels often discourages this. In all the years since I first read the call, this project was the only one I came across that I would be willing to invest four years of my life into. Without you building this project and offering this opportunity to me, in a very literal sense, I would not be who I am today. I am also thankful for your time and patience to change and expand my thinking and training me to use a variety of viewpoints to investigate a problem from different angles. Thank you for your interest and dedication to also discuss minute points, even though in the moment I may not have appreciated it.

I also thank Peter that he, for me and many PhD students with me and before me, has created an incredible scientific space for us to experiment, question, and expand ourselves. A space that provides constant stimulation while giving us the freedom and support necessary to grow. But I also thank him for his personal interest and involvement especially towards the later stage of the PhD project to make this thesis a reality. I also thank my other scientific collaborators, especially Midas and Renato, for their patience and interest to discuss and teach me and their willingness to listen to me even when I didn't necessarily know what I was talking about.

Beyond the scientific and technical issues, completing a PhD project is an emotional and psychological roller-coaster and for me, without Dick, I don't know how I could have dealt with any of those problems. We didn't meet under conditions facilitative of a good relationship and with anybody else, this may have spelled struggle and disaster. But thanks to his generous personality and his honest and supportive attitude he pushed me and helped me, served as a necessary filter when wishful thinking overcame evidence and was just generally indispensable for pretty much every step forward in an uncountable number of occasions. Thank you so much!

Of course, beyond the scientific collaborations, there were many people that supported me emotionally and socially to progress with my PhD. Thank you, Sonia and Chara, for having an open ear and advice when things were most difficult. Thank you, René, Valeria, and Rowan for making my office space a second home which made it worth going to work every day. Also, thank you, Franzi, for welcoming me and tying me to the rest of the group and PhD students that I felt so disconnected from both by topic and experience. And of course, I am thankful for all the wonderful group members I had the pleasure to encounter and interact with every day. Sophie, Ksenija, Ying Ying, Julia, Karin, Iris, Evelyn, Lotte, Dan, and all the NBL members. You made sure I ate regularly, you helped me exercise and you gave me a great space to try and experiment. Thanks to all of you!

The scientific journey to a PhD defense starts long before the actual PhD. I am thankful to Iris, Johan, and Todd for opening my mind to applying formal thinking in areas that continuously try to evade formal treatment and expanding my thinking more than they probably realize. And I thank Raoul and Dominik that allowed me to transition to thinking about the brain while still remaining in my back-then comfortable home of physics.

∞

And finally, I have been so incredibly lucky to have a person who did all of the above for me at the same time – and so much more. Somebody who challenged and educated me scientifically but also made sure I have a comfortable home to retreat to. Who was a supervisor to me when I needed to get to work and who had an open ear and heart when things were difficult. I know that at times she felt like this project was more stressful for her than it was for me and maybe that's true, if only because I had her to lean on and to help me continue. Thank you, Maria, you are the compass to guide me, the wind to push me and the vessel to carry me. I can't wait to see the places you bring me next!

A.5 Biography

Marvin Uhlmann was born in the former GDR and grew up close to Leipzig where he attended the Wilhelm-Ostwald Gymnasium in Leipzig, a school with a specialized profile in math and sciences. After graduating high school and completing his time in the German civil service, he followed his interests in Quantum Physics and Relativity and chose the Swiss Federal Institute of Technology (ETH) in Zürich to study Physics. During his bachelor studies, he spent a formative exchange semester at the National University of Singapore which broadened his experience scientifically but also personally since a large portion of the time was dedicated to traveling. During his studies, his scientific interest had shifted from studying the specific to understanding the complex. This led to him to choose a bachelor's thesis in Astrophysics which was his first experience of studying complex systems through computer simulations.

For his master's education, he sought out a program unifying his skills in exact sciences with his interest in the complexity of the human mind and found it at the Donders Institute for Brain, Cognition and Behavior in Nijmegen. This program offered a broad range of education within cognitive and brain sciences from theoretical cognitive science via neuroimaging all the way to the studies of neurobiology and neurophysics. In a project with the Computational Cognitive Science group, he investigated the computational power of liquid state machines in a theoretical analysis. For his master's thesis in the Neuroinformatics group, he investigated the computational properties of different classes of spiking neural networks acting as biological instances of liquid state machines. This introduced him to high-performance simulation studies of spiking neural networks.

After attaining his master's diploma, he started a PhD project at the Max Planck Institute for Psycholinguistics that united his diverse set of interests in which he investigated how biological neural networks process language with the use of simulations of spiking neural networks, heavily relying on the theoretical analysis of the biological infrastructure on the one hand, and the cognitive and linguistic boundaries on the other hand. This project touched upon a diverse set of disciplines ranging from linguistics and neuroimaging to neurobiology and physics.

Concluding his research, he decided to leave academia for the time being and relocated to Barcelona to interface with the local start-up community. Now he is co-founding an AI start-up and finds himself in a position requiring as diverse a set of skills as ever before, developing technology and AI algorithms as well as taking on the responsibilities to found and build a company.

A.6 MPI Series in Psycholinguistics

1. The electrophysiology of speaking: Investigations on the time course of semantic, syntactic, and phonological processing. *Miranda van Turennout*
2. The role of the syllable in speech production: Evidence from lexical statistics, metalinguistics, masked priming, and electromagnetic midsagittal articulography. *Niels O. Schiller*
3. Lexical access in the production of ellipsis and pronouns. *Bernadette M. Schmitt*
4. The open-/closed-class distinction in spoken-word recognition. *Alette Haveman*
5. The acquisition of phonetic categories in young infants: A self-organising artificial neural network approach. *Kay Behnke*
6. Gesture and speech production. *Jan-Peter de Ruiter*
7. Comparative intonational phonology: English and German. *Esther Grabe*
8. Finiteness in adult and child German. *Ingeborg Lasser*
9. Language input for word discovery. *Joost van de Weijer*
10. Inherent complement verbs revisited: Towards an understanding of argument structure in Ewe. *James Essegbey*
11. Producing past and plural inflections. *Dirk Janssen*
12. Valence and transitivity in Saliba: An Oceanic language of Papua New Guinea. *Anna Margetts*
13. From speech to words. *Arie van der Lugt*
14. Simple and complex verbs in Jaminjung: A study of event categorisation in an Australian language. *Eva Schultze-Berndt*
15. Interpreting indefinites: An experimental study of children's language comprehension. *Irene Krämer*
16. Language-specific listening: The case of phonetic sequences. *Andrea Weber*
17. Moving eyes and naming objects. *Femke van der Meulen*
18. Analogy in morphology: The selection of linking elements in Dutch compounds. *Andrea Krott*
19. Morphology in speech comprehension. *Kerstin Mauth*
20. Morphological families in the mental lexicon. *Nivja H. de Jong*
21. Fixed expressions and the production of idioms. *Simone A. Sprenger*
22. The grammatical coding of postural semantics in Goemai (a West Chadic language of Nigeria). *Birgit Hellwig*
23. Paradigmatic structures in morphological processing: Computational and cross-linguistic experimental studies. *Fermín Moscoso del Prado Martín*

24. Contextual influences on spoken-word processing: An electrophysiological approach. *Daniëlle van den Brink*
25. Perceptual relevance of prevoicing in Dutch. *Petra M. van Alphen*
26. Syllables in speech production: Effects of syllable preparation and syllable frequency. *Joana Cholin*
27. Producing complex spoken numerals for time and space. *Marjolein Meeuwissen*
28. Morphology in auditory lexical processing: Sensitivity to fine phonetic detail and insensitivity to suffix reduction. *Rachèl J. J. K. Kemps*
29. At the same time...: The expression of simultaneity in learner varieties. *Barbara Schmiedtová*
30. A grammar of Jalonke argument structure. *Friederike Lüpke*
31. Agrammatic comprehension: An electrophysiological approach. *Marlies Wassenaar*
32. The structure and use of shape-based noun classes in Miraña (North West Amazon). *Frank Seifart*
33. Prosodically-conditioned detail in the recognition of spoken words. *Anne Pier Salverda*
34. Phonetic and lexical processing in a second language. *Mirjam Broersma*
35. Retrieving semantic and syntactic word properties. *Oliver Müller*
36. Lexically-guided perceptual learning in speech processing. *Frank Eisner*
37. Sensitivity to detailed acoustic information in word recognition. *Keren B. Shatzman*
38. The relationship between spoken word production and comprehension. *Rebecca Özdemir*
39. Disfluency: Interrupting speech and gesture. *Mandana Seyfeddinipur*
40. The acquisition of phonological structure: Distinguishing contrastive from non-contrastive variation. *Christiane Dietrich*
41. Cognitive cladistics and the relativity of spatial cognition. *Daniel B.M. Haun*
42. The acquisition of auditory categories. *Martijn Goudbeek*
43. Affix reduction in spoken Dutch. *Mark Pluymaekers*
44. Continuous-speech segmentation at the beginning of language acquisition: Electrophysiological evidence. *Valesca Kooijman*
45. Space and iconicity in German Sign Language (DGS). *Pamela Perniss*
46. On the production of morphologically complex words with special attention to effects of frequency. *Heidrun Bien*

47. Crosslinguistic influence in first and second languages: Convergence in speech and gesture. *Amanda Brown*
48. The acquisition of verb compounding in Mandarin Chinese. *Jidong Chen*
49. Phoneme inventories and patterns of speech sound perception. *Anita Wagner*
50. Lexical processing of morphologically complex words: An information-theoretical perspective. *Victor Kuperman*
51. A grammar of Savosavo, a Papuan language of the Solomon Islands. *Claudia Wegener*
52. Prosodic structure in speech production and perception. *Claudia Kuzla*
53. The acquisition of finiteness by Turkish learners of German and Turkish learners of French: Investigating knowledge of forms and functions in production and comprehension. *Sarah Schimke*
54. Studies on intonation and information structure in child and adult German. *Laura de Ruiter*
55. Processing the fine temporal structure of spoken words. *Eva Reinisch*
56. Semantics and (ir)regular inflection in morphological processing. *Wieke Tabak*
57. Processing strongly reduced forms in casual speech. *Susanne Brouwer*
58. Ambiguous pronoun resolution in L1 and L2 German and Dutch. *Miriam Ellert*
59. Lexical interactions in non-native speech comprehension: Evidence from electro-encephalography, eye-tracking, and functional magnetic resonance imaging. *Ian FitzPatrick*
60. Processing casual speech in native and non-native language. *Annelie Tuinman*
61. Split intransitivity in Rotokas, a Papuan language of Bougainville. *Stuart Robinson*
62. Evidentiality and intersubjectivity in Yurakaré: An interactional account. *Sonja Gipper*
63. The influence of information structure on language comprehension: A neurocognitive perspective. *Lin Wang*
64. The meaning and use of ideophones in Siwu. *Mark Dingemanse*
65. The role of acoustic detail and context in the comprehension of reduced pronunciation variants. *Marco van de Ven*
66. Speech reduction in spontaneous French and Spanish. *Francisco Torreira*
67. The relevance of early word recognition: Insights from the infant brain. *Caroline Junge*
68. Adjusting to different speakers: Extrinsic normalization in vowel perception. *Matthias J. Sjerps*

69. Structuring language. Contributions to the neurocognition of syntax. *Katrien R. Segaert*
70. Infants' appreciation of others' mental states in prelinguistic communication: A second person approach to mindreading. *Birgit Knudsen*
71. Gaze behavior in face-to-face interaction. *Federico Rossano*
72. Sign-spatiality in Kata Kolok: how a village sign language of Bali inscribes its signing space. *Conny de Vos*
73. Who is talking? Behavioural and neural evidence for norm-based coding in voice identity learning. *Attila Andics*
74. Lexical processing of foreign-accented speech: Rapid and flexible adaptation. *Marijt Witteman*
75. The use of deictic versus representational gestures in infancy. *Daniel Puccini*
76. Territories of knowledge in Japanese conversation. *Kaoru Hayano*
77. Family and neighbourhood relations in the mental lexicon: A cross-language perspective. *Kimberley Mulder*
78. Contributions of executive control to individual differences in word production. *Zeshu Shao*
79. Hearing speech and seeing speech: Perceptual adjustments in auditory-visual processing. *Patrick van der Zande*
80. High pitches and thick voices: The role of language in space-pitch associations. *Sarah Dolscheid*
81. Seeing what's next: Processing and anticipating language referring to objects. *Joost Rommers*
82. Mental representation and processing of reduced words in casual speech. *Iris Hanique*
83. The many ways listeners adapt to reductions in casual speech. *Katja Poellmann*
84. Contrasting opposite polarity in Germanic and Romance languages: Verum Focus and affirmative particles in native speakers and advanced L2 learners. *Giuseppina Turco*
85. Morphological processing in younger and older people: Evidence for flexible dual-route access. *Jana Reifegerste*
86. Semantic and syntactic constraints on the production of subject-verb agreement. *Alma Veenstra*
87. The acquisition of morphophonological alternations across languages. *Helen Buckler*
88. The evolutionary dynamics of motion event encoding. *Annemarie Verkerk*

89. Rediscovering a forgotten language. *Jiyoun Choi*
90. The road to native listening: Language-general perception, language-specific input. *Sho Tsuji*
91. Infants'understanding of communication as participants and observers. *Gudmundur Bjarki Thorgrímsson*
92. Information structure in Avatime. *Saskia van Putten*
93. Switch reference in Whitesands. *Jeremy Hammond*
94. Machine learning for gesture recognition from videos. *Binyam Gebrekidan Gebre*
95. Acquisition of spatial language by signing and speaking children: a comparison of Turkish sign language (TID) and Turkish. *Beyza Sümer*
96. An ear for pitch: on the effects of experience and aptitude in processing pitch in language and music. *Salomi Savvatia Asaridou*
97. lncrementality and Flexibility in Sentence Production. *Maartje van de Velde*
98. Social learning dynamics in chimpanzees: Reflections on (nonhuman) animal culture. *Edwin van Leeuwen*
99. The request system in Italian interaction. *Giovanni Rossi*
100. Timing turns in conversation: A temporal preparation account. *Lilla Magyari*
101. Assessing birth language memory in young adoptees. *Wencui Zhou*
102. A social and neurobiological approach to pointing in speech and gesture. *David Peeters*
103. Investigating the genetic basis of reading and language skills. *Alessandro Gialluisi*
104. Conversation Electrified: The Electrophysiology of Spoken Speech Act Recognition. *Rósa Signý Gisladottir*
105. Modelling Multimodal Language Processing. *Alastair Smith*
106. Predicting language in different contexts: The nature and limits of mechanisms in anticipatory language processing. *Florian Hintz*
107. Situational variation in non-native communication. *Huib Kouwenhoven*
108. Sustained attention in language production. *Suzanne Jongman*
109. Acoustic reduction in spoken-word processing: Distributional, syntactic, morphosyntatic, and orthographic effects. *Malte Viebahn*
110. Nativeness, dominance, and the flexibility of listening to spoken language. *Laurence Bruggeman*
111. Semantic specificity of perception verbs in Maniq. *Ewelina Wnuk*
112. On the identification of FOXP2 gene enhancers and their role in brain development. *Martin Becker*

113. Events in language and thought: The case of serial verb constructions in Avatime. *Rebecca Defina*
114. Deciphering common and rare genetic effects on reading ability. *Amaia Carrión Castillo*
115. Music and language comprehension in the brain. *Richard Kunert*
116. Comprehending Comprehension: Insights from neuronal oscillations on the neuronal basis of language. *Nietzsche H.L. Lam*
117. The biology of variation in anatomical brain asymmetries. *Tulio Guadalupe*
118. Language processing in a conversation context. *Lotte Schoot*
119. Achieving mutual understanding in Argentine Sign Language. *Elizabeth Manrique*
120. Talking Sense: the behavioural and neural correlates of sound symbolism. *Gwilym Lockwood*
121. Getting under your skin: The role of perspective and simulation of experience in narrative comprehension. *Franziska Hartung*
122. Sensorimotor experience in speech perception. *Will Schuerman*
123. Explorations of beta-band neural oscillations during language comprehension: Sentence processing and beyond. *Ashley Lewis*
124. Influences on the magnitude of syntactic priming. *Evelien Heyselaar*
125. Lapse organization in interaction. *Elliott Hoey*
126. The processing of reduced word pronunciation variants by natives and foreign language learners: Evidence from French casual speech. *Sophie Brand*
127. The neighbors will tell you what to expect: Effects of aging and predictability on language processing. *Cornelia Moers*
128. The role of voice and word order in incremental sentence processing. *Sebastian Sauppe*
129. Learning from the (un)expected: Age and individual differences in statistical learning and perceptual learning in speech. *Thordis Neger*
130. Mental representations of Dutch regular morphologically complex neologisms. *Laura de Vaan*
131. Speech production, perception, and input of simultaneous bilingual preschoolers: Evidence from voice onset time. *Antje Stoehr*
132. A holistic approach to understanding pre-history. *Vishnupriya Kolipakam*
133. Characterization of transcription factors in monogenic disorders of speech and language. *Sara Busquets Estruch*
134. Indirect request comprehension in different contexts. *Johanne Tromp*

135. Envisioning Language - An Exploration of Perceptual Processes in Language Comprehension. *Markus Ostarek*
136. Listening for the WHAT and the HOW: Older adults' processing of semantic and affective information in speech. *Juliane Kirsch*
137. Let the agents do the talking: on the influence of vocal tract anatomy on speech during ontogeny and glossogeny. *Rick Janssen*
138. Age and hearing loss effects on speech processing. *Xaver Koch*
139. Vocabulary knowledge and learning: Individual differences in adult native speakers. *Nina Mainz*
140. The face in face-to-face communication: Signals of understanding and non-understanding. *Paul Hömke*
141. Person reference and interaction in Umpila/Kuuku Ya'u narrative. *Clair Hill*
142. Beyond the language given: The neurobiological infrastructure for pragmatic inferencing. *Jana Bašnáková*
143. From Kawapanan to Shawi: Topics in language variation and change. *Luis Miguel Rojas-Berscia*
144. On the oscillatory dynamics underlying speech-gesture integration in clear and adverse listening conditions. *Linda Drijvers*
145. Understanding temporal overlap between production and comprehension. *Amie Fairs*
146. The role of exemplars in speech comprehension. *Annika Nijveld*
147. A network of interacting proteins disrupted in language-related disorders. *Elliot Sollis*
148. Fast speech can sound slow: Effects of contextual speech rate on word recognition. *Merel Maslowski*
149. Reason-giving in everyday activities. *Julija Baranova*
150. Speech planning in dialogue - Psycholinguistic studies of the timing of turn taking. *Mathias Barthel*
151. The role of neural feedback in language unification: How awareness affects combinatorial processing. *Valeria Mongelli*
152. Exploring social biases in language processing. *Sara Iacozza*
153. Vocal learning in the pale spear-nosed bat, Phyllostomus discolor. *Ella Lattenkamp*
154. The effect of language contact on speech and gesture: The case of Turkish-Dutch bilinguals in the Netherlands. *Elif Zeynep Azar*

155. Language and society: How social pressures shape grammatical structure. *Limor Raviv*
156. The moment in between: Planning speech while listening. *Svetlana-Lito Gerakaki*
157. How speaking fast is like running: Modelling control of speaking rate. *Joe Rodd*
158. The power of context: How linguistic contextual information shapes brain dynamics during sentence processing. *René Terporten*
159. Neurobiological models of sentence processing. *Marvin Uhlmann*