



## Locally distributed spatial navigation in a scale-space model for grid cells

Nicolai Sebastian Waniek

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik  
der Technischen Universität München zur Erlangung des akademischen Grades

**Doktor der Naturwissenschaften (Dr. rer. nat.)**

genehmigten Dissertation.

**Vorsitzender:**

Prof. Dr. Eckehard Steinbach

**Prüfende der Dissertation:**

1. Prof. Dr. Jörg Conradt
2. Prof. Dr.-Ing. habil. Alois Knoll
3. Prof. Dr. Alessandro Treves

Die Dissertation wurde am 12.06.2017 bei der Technischen Universität München  
eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am  
21.06.2018 angenommen.



# Abstract

Goal-directed navigation appears to be crucial for mobile animals and robots. Understanding the involved cortical processes is considered to reveal how higher cognitive functions are computed. However, navigation in the mammalian cortex is not sufficiently well comprehended yet which is demonstrated by the discovery of grid cells, a certain type of neuron with enigmatic hexagonally arranged responses. It is commonly assumed that their responsibility is either path integration or localization. Their true computational purpose remains elusive and controversially debated, though.

A novel theory for grid cells is introduced in this thesis. First, a mathematical framework for sequences and transitions is defined rigorously. Propositional logic is used to evaluate characteristics of the framework and it is proved that a hexagonal arrangement of transition encoders is optimal in two dimensional metric space. These results are used to derive the error function of a single grid cell, which also leads to a network model of competitive cells. Afterwards, the algorithmic interactions between grid and place cells, another type of neuron which is relevant for spatial navigation, are considered. For this purpose, a behaviorally significant computational issue when using only a single scale of grid cells, i.e. grid cells which encode transition between two consecutive locations, is pointed out. An extended model which uses only simple yet fundamental algorithms from computer science is introduced to address the problem. In this model, grid cells combine transitions across several spatio-temporal locations and thereby form a distributed scale-space representation of transitions. The cells operate only on locally available information and can be evaluated in parallel. It is shown that the scale-space model improves computational times exponentially and is optimal for a scale increment of  $\sqrt{2}$  between consecutive scales. Furthermore, the connection to other areas of research as well as the relationship between the algorithmic constraints and measurements in biological networks are discussed. For instance, the scale-space model requires temporal buffering which is considered to be related to Theta phase precession, a temporal effect which can be observed both in place and grid cells. Finally, the novel concepts are used in a theoretical study for distributed path computation in a swarm of robots. The resulting algorithms and data structures have relationship to peer-to-peer systems and are analyzed from the perspective of computational complexity.

To summarize, the work gives an entirely novel perspective on grid cells. Their computational purpose, physical realization, as well as algorithmic interactions are derived by introduction of a mathematical framework. Furthermore, the concepts are transported to technical applications.



# Zusammenfassung

Zielgerichtete Navigation scheint essentiell für sich bewegende Tiere und Roboter. Es wird davon ausgegangen, dass ein Verständnis der beteiligten kortikalen Abläufe aufdecken wird wie höhere kognitive Funktionen berechnet werden. Die genauen Berechnungsschritte von Navigation im Hirn von Säugetieren sind jedoch noch nicht hinreichend ergründet, was insbesondere am Beispiel der sogenannten Gitterzellen offensichtlich wird, einem bestimmten Neuronentyp mit rätselhaft anmutendem und hexagonal angeordnetem Antwortverhalten. Üblicherweise wird angenommen dass die Zellen zur Pfadintegration oder aber zur Lokalisation beitragen. Ihre tatsächliche Aufgabe ist jedoch weiterhin nur eingeschränkt erfasst und stark umstritten.

In dieser Arbeit wird eine neue Theorie für Gitterzellen präsentiert. Hierfür wird zuerst ein mathematisches System für Sequenzen und deren Übergänge rigoros definiert. Mit Hilfe von Aussagenlogik werden Charakteristika des Systems erörtert und es wird mathematisch bewiesen, dass eine hexagonale Anordnung von Kodierern von Übergängen optimal in zweidimensionalen metrischen Räumen ist. Diese Ergebnisse werden genutzt um die Fehlerfunktion einer einzelnen Gitterzelle herzuleiten, was daraufhin zu einem Netzwerk an kompetitiven Zellen führt. Anschließend werden die Interaktionen zwischen Gitter- und Ortszellen, einem weiteren Typ Neuron der notwendig für räumliche Navigation ist, algorithmisch betrachtet. Zunächst wird ein bedeutendes rechentechnisches Problem dargelegt wenn lediglich eine Auflösung von Gitterzellen verwendet wird, also Zellen, die ausschließlich Übergänge zwischen aufeinanderfolgende Orten lernen. Daraufhin wird ein erweitertes Modell vorgestellt welches lediglich einfache, aber fundamentale, Algorithmen der Informatik nutzt um das Problem zu beheben. Gitterzellen kombinieren Übergänge aus mehreren raum-zeitlichen Auflösungen innerhalb des neuen Modells und erzeugen dadurch eine verteilte Skalenraum-Repräsentation von Übergängen. Hierbei agieren die Zellen lediglich auf Grund von lokal verfügbaren Informationen und ihre Auswertung kann parallelisiert stattfinden. Es wird dargelegt dass die Skalenraum-Repräsentation Berechnungszeiten exponentiell beschleunigt und dass die Auflösung von einer Skala zur nächsten optimal um den Faktor  $\sqrt{2}$  vergrößert wird. Anschließend wird die Verbindung zu anderen Forschungsrichtungen ergründet und ebenso die algorithmischen Notwendigkeiten in Bezug auf Messungen von biologischen Zellen diskutiert. Zum Beispiel benötigt das Skalenraum-Modell zeitliche Pufferung von Daten. Dies ist verwandt zur Phasenpräzession bezüglich des Theta-Rhythmus. Letzteres ist ein zeitlicher Effekt der sowohl in Orts- als auch Gitterzellen messbar ist. Schließlich werden die neuen Konzepte in einer theoretischen Studie für verteilte Pfadberechnung in einem Schwarm von Robotern verwendet. Die daraus resultierenden Algorithmen und Datenstrukturen besitzen Ähnlichkeit zu *Peer-to-Peer* Netzen und werden aus

Sicht der Berechnungskomplexität analysiert.

Zusammengefasst präsentiert die Arbeit eine gänzlich neue Betrachtungsweise auf Gitterzellen. Durch die Einführung eines mathematischen Systems kann sowohl deren Aufgabe in Berechnungen, ihre physikalische Realisierung, als auch die algorithmischen Interaktionen erörtert werden. Außerdem werden die neuen Konzepte auf technische Anwendungen übertragen.

# Preface

Spatial navigation is mesmerizing. How can creatures as tiny as ants or as complex as humans find their goals by integrating their sensory stimuli? And what makes them different to each other and to technical systems and virtual agents? Most of the latter either require massive computing power or simply fail in generating compelling trajectories despite their complex algorithms. Is their complexity a burden rather than a benefit? Can an understanding of how the brain computes spatial navigation enhance robots and distributed technical systems? Little did I know how involved the subject is.

Thanks to Heiko Neumann and Florian Raudies I came into contact with cortical computations, spatial navigation, and eventually grid cells while I was still an undergraduate student at Ulm University. And I was intrigued ever since. How is the hexagonal pattern that grid cells generate of any computational utility during navigation?

I could finally spend all of my time to follow up on this question after joining Jörg Conradt's group Neuroscientific System Theory (NST) at the Technische Universität München by the end of 2012. I could freely explore the subject from the perspectives of a neuromorphic engineer, a computer scientist, as well as a computational neuroscientist only because he provided the ideal environment for this purpose. He was supportive and provided guidance with the pragmatic view of an engineer even when I came with yet another uncommon idea. Furthermore he got me in touch with Edvard Moser, Alessandro Treves, and Richard Morris, all who provided precious insights to understand how spatial information is processed in the brain and how this leads to behavior.

Shortly afterwards I started to see hexagons everywhere. However, I also became increasingly frustrated with the existing models of grid cells. They answered the question how the phenomenological response of grid cells could be generated, but did not address their computational advantage in a way which satisfied me. As a computer scientist and having written software for many years, I was trained to formalize the core of a problem before starting to write a solution. Hence, I took a step back and tried to apply the same approach to grid cells and spatial navigation. Inspired by the quote "Make things as simple as possible, but not simpler", commonly attributed to Albert Einstein, I searched for the easiest mathematical concept in which grid cells had a distinct purpose and formed hexagonal response fields. In the end, this led to the development of the novel theory and subsequently also the model of the entorhinal-hippocampal loop, both of which are proposed in this thesis. In the proper sense of the quote, only relatively simple but nevertheless powerful and well-known algorithms were used in the development of the model.

Consequently, this thesis is a journey from abstract symbolic logic and mathematics, crosses the landscape of computational neuroscience, and finally reaches the realm of algorithms. Certainly I do not expect a reader to be familiar with all of these subjects. It took several years of hard work to discover the connections between all of them. For me, that is. Hence, a rather generic introduction to neural modelling is given, a condensed overview on the current knowledge about how the brain represents spatial information for the purpose of navigation is presented, and the formalisms and descriptions of algorithms are kept as straightforward as I thought possible.

This work would not have been possible without the continued support by my family and friends. They made sure that I kept a balanced life and brought joy throughout the entire ride. I'm especially grateful to my parents and Maïke, who bore with me even when things did not go as planned.

Nicolai Waniek  
Munich, June 2017

# Table of Contents

Abstract	i
Zusammenfassung	iii
Preface	v
List of Acronyms	xi
List of Figures	xiii
List of Tables	xiv

## Part I Foundations

<b>1 Overture</b>	<b>3</b>
1.1 Fundamental philosophy and motivation . . . . .	3
1.2 Research questions and scope of the thesis . . . . .	7
1.3 Organization of the thesis . . . . .	8
1.4 Contributions to and of the thesis . . . . .	10
1.4.1 List of Publications . . . . .	11
<b>2 Biological and artificial neural networks</b>	<b>13</b>
2.1 Neurons and synapses . . . . .	13
2.2 Neural networks and associative memories . . . . .	15
2.3 Plasticity, synchronization, and learning . . . . .	17
2.4 Modularity and hierarchical computation . . . . .	19
2.5 On the employed modelling approach . . . . .	20
<b>3 The neural representation of space</b>	<b>21</b>
3.1 The Hippocampal Formation and Entorhinal Cortex . . . . .	21
3.2 A zoo of spatially modulated neurons . . . . .	24
3.2.1 Place cells . . . . .	24
3.2.2 Head direction cells . . . . .	26
3.2.3 Grid cells . . . . .	27
3.2.4 Boundary vector cells / border cells . . . . .	29
3.2.5 Linear speed cells . . . . .	30
3.2.6 Conjunctive cells . . . . .	30
3.2.7 Interneurons . . . . .	31
3.3 Neural interactions and concluding remarks . . . . .	31

## Part II Multi-Transition Theory with an application to Neural Spatial Navigation

<b>4</b>	<b>The motivation for a novel theory</b>	<b>35</b>
4.1	A brief tour of models for grid cells . . . . .	35
4.2	Core question and related work . . . . .	37
<b>5</b>	<b>On Multi-Transition Systems</b>	<b>39</b>
5.1	Alphabets and the computational logic of path planning . . . . .	39
5.2	Universal Multi-Transition Systems . . . . .	40
5.2.1	Interim observations and implications for neural networks . .	44
5.3	Sequences in continuous metric space: Emergence of grid cells . . . .	45
5.3.1	On dense sampling and sphere packing . . . . .	45
5.3.2	Spatial neighborhood transitions and grid cells . . . . .	45
5.4	Discussion and remarks on the biological plausibility . . . . .	47
<b>6</b>	<b>A neural model of self-organizing grid cells</b>	<b>51</b>
6.1	Learning to decorrelate input and output . . . . .	51
6.1.1	Model overview and implementation details . . . . .	51
6.1.2	Simulation results and discussion . . . . .	55
6.2	Single neuron model of a grid cell . . . . .	55
6.2.1	On dendritic tree computation and the error function of a single transition neuron . . . . .	57
6.2.2	Model characterization and simulation results . . . . .	60
6.2.3	Interpretation of the model and results . . . . .	61
6.3	Competitive network model of grid cells . . . . .	65
6.3.1	Model description and network dynamics . . . . .	66
6.3.2	Methods and simulation results . . . . .	68
6.3.3	Discussion of the model and its results, predictions, and future work . . . . .	71
<b>7</b>	<b>Algorithmic exploration of the entorhinal-hippocampal loop</b>	<b>75</b>
7.1	The Universal Multi-Transition System as Growing Neural Gas . . .	76
7.1.1	Model and implementation details . . . . .	76
7.1.2	Simulation and results . . . . .	78
7.1.3	Brief discussion of the temporal transition system . . . . .	79
7.2	A scale-space model for spatial navigation . . . . .	80
7.2.1	Multiple scales and the algebraic number $\sqrt{2}$ . . . . .	83
7.2.2	Simplified model and results . . . . .	86
7.3	Discussion, observations, predictions . . . . .	89
7.3.1	Temporal buffering, Theta phase precession, and number of scales . . . . .	93
7.3.2	On the expected number of neurons per scale . . . . .	96
7.3.3	Relationship to algorithms and concepts from computer science	98
7.4	Conclusion and future work . . . . .	98

## Part III Beyond MTT and Neural Spatial Navigation

<b>8</b>	<b>Towards massively distributed spatial navigation</b>	<b>103</b>
8.1	Problem formulation and related work . . . . .	104
8.2	Algorithms and Data Structures . . . . .	105
8.2.1	Definitions for single and multiple layers . . . . .	105
8.2.2	Construction algorithms for the data structures . . . . .	107
8.2.3	Algorithms for retrieval . . . . .	109
8.3	Discussion and future work . . . . .	110
<b>9</b>	<b>Concluding remarks and potential directions</b>	<b>113</b>
	<b>Appendices</b>	<b>114</b>
<b>A</b>	<b>Partial derivatives of the error function of the single grid cell model</b>	<b>117</b>
<b>B</b>	<b>Calculation of gridness score, grid orientation, and orientation error</b>	<b>119</b>
<b>C</b>	<b>Preliminary results in favor of the dense sampling assumption</b>	<b>121</b>
<b>D</b>	<b>Algorithms for Sparse Layered Graphs and Transition Graphs</b>	<b>125</b>
<b>E</b>	<b>On the complexity of algorithms for SLGs and TGs</b>	<b>127</b>
E.1	Construction algorithms . . . . .	127
E.2	Retrieval algorithms . . . . .	130
	<b>Bibliography</b>	<b>131</b>



# List of Acronyms

<b>AI</b> Artificial Intelligence . . . . .	3
<b>ANN</b> Artificial Neural Network . . . . .	4
<b>BCM</b> Bienenstock-Cooper-Munro . . . . .	17
<b>CA1</b> Cornu Ammonis 1 . . . . .	21
<b>CA2</b> Cornu Ammonis 2 . . . . .	21
<b>CA3</b> Cornu Ammonis 3 . . . . .	21
<b>CAN</b> Continuous Attractor Neural Network . . . . .	16
<b>CMOS</b> Complementary Metal-Oxide-Semiconductor . . . . .	4
<b>CNN</b> Convolutional Neural Network . . . . .	4
<b>CSP</b> Communicating Sequential Processes . . . . .	10
<b>DG</b> Dentate Gyrus . . . . .	21
<b>DNN</b> Deep Neural Network . . . . .	4
<b>EC</b> Entorhinal Cortex . . . . .	21
<b>EEG</b> Electroencephalogram . . . . .	18
<b>GNG</b> Growing Neural Gas . . . . .	9
<b>GPGPU</b> General-Purpose Computing on Graphics Processing Units . . . . .	4
<b>GPU</b> Graphics Processing Unit . . . . .	4
<b>HD</b> Head Direction . . . . .	26
<b>HEPV</b> Hierarchical Encoded Path Views . . . . .	105
<b>HF</b> Hippocampal Formation . . . . .	21
<b>HiTi</b> Hierarchical Performance Multi-Level Routing . . . . .	105
<b>LDA</b> Locally Distributed Algorithm . . . . .	5
<b>IEC</b> Lateral Entorhinal Cortex . . . . .	22

<b>LFP</b> Local Field Potential . . . . .	18
<b>LIF</b> Leaky-Integrate and Fire . . . . .	73
<b>LoG</b> Laplacian of Gaussian . . . . .	84
<b>LTP</b> Long-Term Plasticity . . . . .	18
<b>mEC</b> Medial Entorhinal Cortex . . . . .	22
<b>MTS</b> Multi-Transition System . . . . .	9
<b>MTT</b> Multi-Transition Theory . . . . .	9
<b>OOP</b> Object-Oriented Programming . . . . .	83
<b>pdf</b> Probability Density Function . . . . .	14
<b>PFC</b> Pre-Frontal Cortex . . . . .	22
<b>RDS</b> Reaction-Diffusion System . . . . .	61
<b>RL</b> Reinforcement Learning . . . . .	19
<b>SIFT</b> Scale-Invariant Feature Transform . . . . .	98
<b>SLAM</b> Simultaneous Localization and Mapping . . . . .	103
<b>SLG</b> Sparse Layered Graph . . . . .	105
<b>SNC</b> substantia nigra pars compacta . . . . .	82
<b>STDP</b> Spike-Timing Dependent Plasticity . . . . .	9
<b>STP</b> Short-Term Plasticity . . . . .	18
<b>SURF</b> Speeded-Up Robust Features . . . . .	98
<b>SWP-R</b> Sharp Waves and Ripples . . . . .	26
<b>TG</b> Transition Graph . . . . .	105
<b>VLSI</b> Very-Large-Scale Integration . . . . .	4

# List of Figures

1.1	Place and grid cell firing fields of single cells . . . . .	6
2.1	Hand-drawing of a neuron, abstraction, and different types of connectivity . . . . .	14
3.1	Schematic of areas in the rodent brain and exemplary spike responses of place and grid cells . . . . .	23
3.2	Theta and Theta phase precession . . . . .	25
3.3	Head direction cell firing follows distal cues . . . . .	27
3.4	Properties of grid cell firing fields . . . . .	28
3.5	Place and grid cell remapping . . . . .	28
3.6	Boundary vector cells . . . . .	30
5.1	Example of symbols and transitions as a bipartite graph and its reduction to a undirected graph . . . . .	44
5.2	Spatial symbol and densest packing of symbols in two dimensions . . . . .	46
5.3	Transition graph in a metric space and its graph coloring . . . . .	46
6.1	STDP decorrelation hypothesis . . . . .	52
6.2	STDP kernel and weight distribution and model network layout . . . . .	53
6.3	Evolution of weights in feed-forward (FF) and recurrent feedback (FB) projections and examples at end of simulation . . . . .	57
6.4	Overviews of the single- and multi-cell models . . . . .	60
6.5	Single grid cell model, gridness scores over receptive field sizes . . . . .	62
6.6	Single grid cell model, gridness scores over error importances . . . . .	63
6.7	Single grid cell model, state of convergence after 5000 iterations for varying $\sigma_1$ . . . . .	64
6.8	Trajectory example and movement statistics . . . . .	68
6.9	Competitive grid cell network, gridness score over time (400 simulations)	69
6.10	Competitive grid cell network, relative orientation error over time (400 simulations) . . . . .	69
6.11	Examples for the weight evolution and final auto-correlation maps . . . . .	71
7.1	(Temporal) transition model . . . . .	77
7.2	Algorithms for learning and retrieval in a temporal transition system $\mathcal{M}$	78
7.3	S-shaped trajectory and replay . . . . .	79
7.4	Pre-play activity within the network after exploration learning of additional transitions at several time steps . . . . .	79

7.5	Combined spatio-temporal transition model . . . . .	81
7.6	Example for temporal, spatial, and spatio-temporal transition computation. . . . .	82
7.7	Scale space construction and multi-scale response fields . . . . .	85
7.7	Algorithm for learning transitions in a spatio-temporal transition system $\mathcal{P}$ . . . . .	87
7.8	Algorithm for retrieval of transitions in a spatio-temporal transition system $\mathcal{P}$ . . . . .	88
7.9	Computational times of the scale-space model . . . . .	89
7.10	Functional levels of goal-directed navigation and localization . . . . .	94
7.11	Loop nesting and Theta phase precession . . . . .	95
7.12	Expected number of grid cells per scale . . . . .	97
8.1	Graph, cells, and border nodes . . . . .	106
8.2	Edge contraction during the construction of a Sparse Layered Graph (SLG) . . . . .	107
8.3	Construction of a Sparse Layered Graph (SLG) with multiple layers . . . . .	108
8.4	Construction of a Transigion Graph (TG) . . . . .	109
8.5	Example for a shortest path query in SLG and TG . . . . .	110
B.1	Example for the extraction of gridness and orientation . . . . .	120
C.1	Preliminary results of sampling from the boundary vector space . . . . .	122

## List of Tables

6.1	Model parameters for learning receptive fields . . . . .	54
C.1	Parameters for sampling boundary vector. . . . .	122

# **Part I**

## **Foundations**



# Chapter 1

## Overture

### 1.1 Fundamental philosophy and motivation

Autonomous mobile robots are already ubiquitous. On the lower end of the complexity spectrum there are lawn mowers and vacuum cleaners, on the other end self-driving cars and unmanned autonomous aerial vehicles. Despite their differences in complexity and function, each robot is obliged to process significant amounts of sensory data in real-time. Permanent and timely evaluation of streams of information is a prerequisite to safely operate in any surrounding, adapt to a constantly changing world, or interact with other robots or humans cooperatively. However, data processing is exacerbated by malfunctioning sensors, noisy readings, or data transmission errors, to name only the most apparent obstacles. Failure in performance can lead to dramatic consequences. Furthermore, there is increased interest to employ robots in scenarios and environments with elevated levels of intricacy. For instance, robots for elderly care will not only have to cope with cluttered environments and aggravated human-robot interaction [106]. In the case of an emergency, they are also likely to encounter previously unknown mixtures of stimuli but still need to independently deduce life-saving counter-measures. Furthermore, they have to collaborate with other robots and humans. Conclusively, there is significant demand for improved robots and robust techniques for complex settings [98].

The next evolutionary leap forward in robotics is considered to be Artificial Intelligence (AI), in the hope that it is suitable for higher cognitive functionality. Tasks which are seemingly simple for humans such as grasping an object with a hand, goal-directed navigation, or verbal and non-verbal interaction with other humans are hard control problems. Engineered solutions typically require several independent but integrated models, for instance for actuators, feasible parameters spaces, or the surrounding world. Frequently, such models are not available before initial operation or have to be adaptive to account for dynamically changing settings and requirements. Furthermore, the control problems themselves are often mathematically intractable. Thus, hope of many researchers is that AIs, equipped with the capability to learn and adapt to novel situations, will solve many of these difficult control problems. These problems are thought to require higher cognitive abilities to be solved. Unsupervised and autonomous learning is considered to pave the way for artificial systems which can reason on abstraction levels just like humans, or even surpass them. Furthermore, it is expected that novel results of *cognitive computing*, a term used to describe any technology which integrates AI, signal processing, and other related areas, will

express fault tolerance and contextual reasoning at so far unseen levels.

A currently popular candidate to address many of the algorithmic obstacles are Artificial Neural Networks (ANNs). They are thought to be a cornerstone for developing higher cognitive AIs [316]. This comes not as surprise, for the simple reason that ANNs are often inspired by and modelled after real biological neural networks. In fact, ANNs were conceived as model to study the computational properties of real neural systems [238]. They are able to learn generalized representations of data and adapt to fuzzy and novel inputs, making them resilient against data transmission issues [80, 301]. Just recently novel ANN, for ANNs in the form of so-called Deep Neural Networks (DNNs) or Convolutional Neural Networks (CNNs) excelled [128, 206], and outperformed almost any competitor during classification tasks – even humans [129, 194, 326]. However, life-long and continuous learning are still research in progress. Furthermore, unsupervised learning, i.e. learning internal representations and generalization without tremendous amounts of pre-defined training data, is an unsolved issue. Both issues are considered exceptionally important for true AI, though [316].

State of the art to process and train ANNs either utilizes clusters of supercomputers or employs General-Purpose Computing on Graphics Processing Units (GPGPU) [64, 194]. Both approaches are currently questionable for truly autonomous robotics, especially if the robots are required to be small, have to operate in a life-long scenario, or will be deployed in remote areas without access to global information networks. Despite recent advances concerning the power consumption of Graphics Processing Units (GPUs), shortage of battery resources exclude either technique on small robots. The lack of access to networks of information is troublesome for systems which have to express sustained learning to adapt to dynamically changing environments. In addition, the computational principles of neural networks seem to differ significantly from the operations of classical von Neumann architectures, i.e. both memory and code execution are integrated into a single neuron in real biological tissue.

Lately, a novel branch of engineering seeks to address the limitations of physical size and power consumption. Going by the name *neuromorphic computing*, this interdisciplinary research area combines engineering, mathematics, computer science, and computational neuroscience to simulate large neural networks. With the help of Very-Large-Scale Integration (VLSI) that uses analog hardware or integrates analog and digital hardware in a mixed-mode, engineers develop systems and sensors which are inspired by biological neural networks. In neuromorphic systems, neurons mimic both the function and morphology of their biological counter-parts. Realization in Complementary Metal-Oxide-Semiconductor (CMOS) or, more recently, memristor technology promises ultra-low power at exceptionally high speeds of processing in parallel and distributed modes [172, 274]. Akin biological neural networks, a typical neuromorphic chip consists of many simple processing entities which are massively interconnected. The recent availability of this novel kind of hardware, e.g. [101, 160, 268], allows to run significantly larger biologically plausible neural networks than before [35], or realize neural principles in real-time scenarios and on robotic platforms [389].

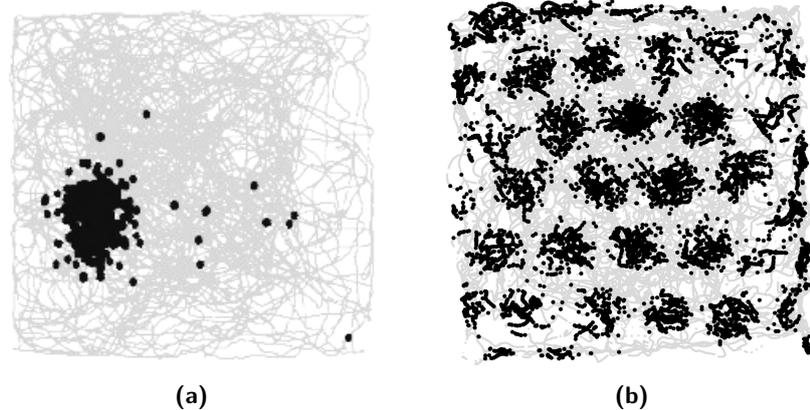
It is essential to recognize that neural networks are independent distributed processes that operate exclusively on local information without access to a global

state [305, 306]. Thereby they execute what is called a *Locally Distributed Algorithm (LDA)*. Only the concerted operation and concurrent activation of multiple neural entities, including synchronization mechanisms and message passed data exchange, yields a global result. Hence it seems natural to apply methods from computer science, e.g. tools for the analysis and description of message-passed distributed systems, to understand biological neural networks. Especially higher cognitive functions involve multiple distal cortical areas, transitions between internal states of these areas, and intra- as well as inter-area data exchange. Precisely this approach of understanding, which is rooted in concepts of computer science, is used in Part II of this thesis. There, goal-directed navigation is analyzed as a distributed system using a completely novel theory based on transitions.

Understanding cognitive functionality on a conceptual level may be fruitful to advance AI. In neuroscience, spatial navigation is viewed as a model for higher cognitive operations because it involves the retrieval and integration of memories and multiple sensory stimuli as well as planning [59]. The behavior during as well as the cortical areas responsible for spatial navigation in rodents are exceptionally well studied. In fact, research and literature date back already over a century and were concretized when Tolman proposed the cognitive map theory in 1948 [354]. In this theory, he suggested that animals and humans not only learn a simple stimulus-response function for the purpose of goal-directed navigation, but acquire an internal map of their surrounding. Due to subsequent decades of research, huge amounts of electrophysiological recording data are available nowadays. Nevertheless, research on the neural representation of space continues with important novel discoveries, hypotheses, and theories published almost weekly. Awarding John O’Keefe and May-Britt and Edvard Moser with the Nobel Prize in 2014 for their outstanding rigorous work is just the most prominent recent climax. Their discovery of certain neurons involved in spatial navigation, namely place and grid cells [135, 264, 265], lead to a novel perspective on goal-directed navigation. However, spatial navigation in the rodent brain is not conclusively understood due to surprising characteristics of several involved neurons as well as the lack of a coherent understanding of their interactions.

Grid cells for instance, besides several other puzzling properties, demonstrate peculiar response patterns [135]. To anticipate Part II, their grid fields, i.e. the activity of a single neuron with respect to locations in the environment, appear to be distributed hexagonally and form a regular tessellation of space (Figure 1.1b). Several competing hypotheses about the origin of the hexagonal grid pattern exist, but so far there is no distinct favorite of any of the models among researchers. With the help of tools from computer science, specifically distributed system analysis, a novel theory for grid cells is derived in this thesis, which in turn is used to develop a self-organized model for grid cells.

Another property of grid cells is equally astonishing. It was observed that the sizes of grid fields of different grid cells vary in discrete steps [336]. Furthermore, the scale factor between these steps is approximately constant with a value suspiciously close to  $\sqrt{2}$  [336]. Although several hypotheses were put forward, the mechanism for the self-organization and, ultimately, the computational purpose for discrete grid scales remain elusive. Analyses were published that describe the purpose from mathematical perspectives, primarily multi-scale probabilistic inference and multi-



**Figure 1.1 – Place and grid cell firing fields of single cells.** (a) Black dots indicate spiking activity of a single place cell with respect to a square environment. The gray line shows the trajectory that the animal was walking. (adapted with permission from [84]). (b) Each black dot represents spiking activity of a single grid cell during explorative movement of an animal. The gray lines indicate the trajectory that the animal was walking. The response of the cell tessellates the environment in an almost perfect hexagonal grid. (reproduced with permission from [256])

resolution analysis [335,375]. Unfortunately none of the models convincingly explains the emergence of the discretization. Another issue with existing models is that they leave an inconvenient issue, called *problem of double redundancy* in the scope of this thesis, completely unaddressed. Place cells differ from grid cells in that they express only singular or very few fields of activity with respect to an environment (Figure 1.1a). Therefore it is likely that place cells are encoders of single locations. However, it was mathematically demonstrated that the grid cell system itself, if it also were to encode spatial locations, tremendously surpasses accuracy and fault-tolerance of the place cell system [334,335]. Given the fact that biological neural networks are energetically expensive to maintain, the question arises why there should be two systems, which are replicated in each hemisphere of the brain, performing the same computation. Furthermore, most models for the interaction of place and grid cells ignore temporal and episodic aspects, also encoded in the areas in which place cells were found [107,319,358,364]. Hence it appears that there is something fundamental missing in the current understanding of individual functions of grid and place cells, in their cooperative interactions, and their organization. Part II addresses these issues by deriving a novel theory for grid cells and observing their algorithmic properties.

Finally the question remains if knowledge about the principles of neural spatial navigation is useful for rather technical scenarios. To explore a potential answer for this question, one specific artificial use-case is studied in Part III of the thesis. There, it is assumed that a massive swarm of robots needs to cooperatively find a way from a start to a goal for one of its participating agents. However, each robot is limited in its knowledge about the environment. Therefore, principles of Part II are applied to derive a fully distributed map for spatial navigation on multiple participating hosts.

Summarizing this overture, there is high demand for novel algorithms to advance AI towards higher cognitive functions, inspired by biological systems. Ideally, these algorithms will run on neuromorphic hardware and express self-organization capabilities to account for dynamically changing demands. Furthermore, they should

operate distributively. However, one of the primary cognitive functions in mammals, namely goal-directed spatial navigation, is not completely unravelled yet. Working towards a conclusive understanding of spatial navigation in the rodent brain is thus doubly relevant. On the one hand such knowledge is likely to provide access to understand other higher cortical operations. On the other hand it is perceived helpful in improving current AI. It is necessary to answer several very specific research questions for this purpose.

## 1.2 Research questions and scope of the thesis

The main focus of this thesis is to understand the functional interactions of the primary areas involved in spatial representation and navigation in the rodent brain. These insights are subsequently used to extract algorithms for massively distributed settings. Given the considerations of the previous section, the following concrete questions can be stated.

1. How is space represented in the rodent brain?

In other words, what are the known neuron types that are involved in representing spatial information. The question is addressed in form of an overview of relevant neurons in the rodent brain.

2. How is information processing for spatial navigation split into and distributed across modalities in the brain?

A reasonable approach to understand the information processing performed during spatial navigation is to look at neurophysiological data. It is possible to derive connectivities and thereby the flow of information between cortical areas using these data. Combined with single neuron recordings it is possible to assign specific functionality to certain areas. Usually, these observations are used to model specific neural responses. In addition, recordings can be used to reason about functional properties of individual cells.

In this thesis the question is answered from a different perspective. First, a novel formalism is defined which starts from a purely theoretical point of view for goal-directed navigation. The formalism is developed with a focus on optimality. Here, grid cells emerge as a theoretically optimal encoder of spatial transitions. The theoretical results are subsequently used to derive and simulate biologically plausible neural models.

3. What are the consequences of the formalism, i.e. encoding of transitions, with respect to computational performance?

One of the behaviorally relevant tasks of animals is to compute trajectories to goal-locations. Certainly, the computation should be performed as quickly as possible to avoid severe consequences due to predators. On the other hand, mechanisms for short-cut finding are desirable to minimize energy consumption. These points are addressed from an algorithmic point of view, in which a novel *scale-space model* for goal-directed navigation in the entorhinal-hippocampal loop is proposed. The suggested algorithms operate only on locally available

information and express parallelism. The results of the model are observed theoretically as well as using simulations.

4. What predictions can be made?

The novel formalism and concept lead to concrete predictions. For instance, the model allows to discuss interactions between place and grid cells on the synaptic level. Furthermore, the algorithmic model requires temporal buffering. A candidate for such a buffering mechanism was observed in the real rat. Besides other connections, the link between these observations and the model will be discussed in detail.

5. Is it possible to extract the proposed principles of path computation in the rodent brain and transfer them to technical systems?

The basic formalism treats goal-directed navigation with only a single animal in mind. However, robots will have to express increased levels of cooperation in the future. Therefore the novel concepts of spatial navigation were applied to a swarm of robots. The complexities of the resulting algorithms are examined theoretically.

### 1.3 Organization of the thesis

The first part of the thesis begins, unsurprisingly, with Chapter 1. It is an "overture" that contains the basic perspective and motivation on the subject of goal-directed navigation, and why it is relevant. The arising questions during this introduction are subsequently concretized. Right now you are reading an overview of the structure and summary of the thesis. The chapter closes with a statement about the contribution, i.e. an overview of publications, and impact of other people to the work presented in the thesis.

Foundations of artificial as well as biological neural network theory are given in Chapter 2. Starting from single neurons and synapses, the chapter will expand to associative memories and learning in networks of neurons. Furthermore, it is addressed how neural networks in real cortices are separated by functionality. Thereby they often form hierarchies and levels of computation. The chapter closes with a statement about and motivation of the modelling approach taken in the thesis.

Chapter 3 introduces the areas of interest in the rodent brain during goal-directed navigation. First, they are discussed with respect to their connectivity. Afterwards, descriptions and characterizations of relevant neurons found in these areas are given. It is also presented how some of these neurons can be characterized or modelled. Furthermore, interactions between several types of neurons are discussed. Issues with existing models for these neurons, especially grid cells, are remarked at the end of the chapter, which concludes the first part of the thesis.

The second and main part of the thesis opens with Chapter 4. The necessity of a novel theory and model for grid cells, or goal-directed navigation in general, is motivated after a brief tour of existing models. An important issue which afflicts most of the models for grid cells is addressed and given the name *problem of double redundancy*. Finally, a summary of part two of the thesis is presented.

A novel theory called Multi-Transition Theory (MTT) and formalisms for Multi-Transition System (MTS) are developed in Chapter 5. In the chapter, the theory is deliberately kept abstract. Thereby, it can be applied to other domains than just goal-directed navigation. Nevertheless, the formalisms are presented in the context of spatial navigation. First, goal-directed navigation is described in terms of sequences on symbols of alphabets. Then, it is asked how to optimally encode arbitrary transitions between symbols by which the concept of a *transition bundle* is introduced. Afterwards, the formalisms are extended to metric space. Here, it is proved that a hexagonal arrangement of a finite number of transition encoders can represent transitions in an infinite space. The chapter closes with remarks on the biological plausibility of the formalism, and necessary consequences for neural networks.

Chapter 6 first addresses an important prerequisite of the theoretical results of Chapter 5 in neural networks. Namely, *transition bundles* require the capability to decorrelate from input states and correlate to target states and thereby forming center-surround receptive fields. This behavior is analyzed in a simple neural network which uses plastic connectivity that is governed by Spike-Timing Dependent Plasticity (STDP). Simulations show the formation of on-center off-surround receptive fields. These results are used afterwards to derive the error function of a single grid cell. Finally, these findings are extended to a competitive network of grid cells. They self-organize while a simulated agent travels through an environment.

After demonstrating biological plausibility in the previous chapter, Chapter 7 addresses the interactions between grid cells, place cells, and sensory representations. First, a temporal transition system is presented for this purpose. The transition system uses Growing Neural Gas (GNG) for recruitment of novel neurons. Then, a spatial transition system is incorporated. The design decision to separate spatial and temporal transitions is discussed both from a biological as well as a computer scientific perspective. Afterwards, issues with respect to computational performance are noted and addressed. To solve these issues, a novel *scale-space* model for spatial navigation is presented. In the model, multiple discrete scales of grid cells with a scale increment of  $\sqrt{2}$  emerge as the optimal solution for spatial look-ahead. The model is then further simplified for simulation purposes which demonstrate an exponential speed-up of computational times. Subsequently, a detailed discussion about the biological plausibility, predictions with respect to the entorhinal-hippocampal loop, and relations to algorithms from computer science are presented. The chapter closes with a short discussion of ongoing and future work on the model, and hereby concludes the second part of the thesis.

The third and final part of the thesis opens by moving concepts presented in the previous chapters, in particular multi-scale clustering and transitions, towards a technical application. The problem of spatial navigation and mapping in a network of cooperatively operating robots is considered Chapter 8 in form of a *theoretical assessment*. Here, two novel multi-layer data structures, and the algorithms to construct them, are introduced. The data structures are defined in a way which allows to distribute intermediate parts onto an arbitrary number of participating hosts. The algorithms for the construction of the data structures can be parallelized in several intermediate steps. Furthermore, shortest distance and shortest path computation are described.

Chapter 9 finally concludes the thesis. It re-examines the findings presented in the previous parts and chapters, and points to future work.

## 1.4 Contributions to and of the thesis

This dissertation was partially supported by the European Union grant GRIDMAP, Future in Emerging Technologies (FET) project 600725, funded under Framework Programme 7 "Information and communication technologies" (FP7-ICT). Several valuable discussions with participants and examiners of the project lead to the development of the novel concepts. Most notably were meetings in person and written communications with Alessandro Treves to help improve the understanding of existing models for grid cells, especially his rate adaptation model [196]. Furthermore, discussions with Richard Morris and Edvard Moser lead to the realization that the grid cell system is supportive but not sufficient for behavior expressed by rodents during spatial navigation. They also helped to clarify the interactions between several areas of the entorhinal-hippocampal loop on a synaptic level. Additional important insight into transition systems and how they can be used in terms of neural modelling were gained during discussions with Philippe Gaussier, who is one of the authors of a transition model of the entorhinal-hippocampal loop which is closely related to the model presented in this thesis [73, 74, 146].

The overviews of neural networks, modelling, and the representation of space in the rodent brain, presented in the final two chapters of Part I, are reviews of research conducted by others. Particularly influential work is clearly stated in these chapters, for instance when discussing the modelling approach adopted in the thesis, which was inspired by David Marr [230].

The thesis contributes an entirely novel perspective on grid cells in Part II in which it is proposed that grid cells form an optimal encoding of a multi-transition system. This perspective and the associated formalisms, models, and simulations presented in that part of the thesis were derived and developed by the author of this thesis. Note however that transition systems as such are a well-known formal concept from computer science to examine automata [345]. In addition, this thesis combines the logic of transition systems with notations used by Tony Hoare in his formulation of Communicating Sequential Processes (CSP) [147]. Furthermore, the temporal interpretation of events in a neural system was inspired by the analysis of time in distributed systems, introduced primarily by Leslie Lamport [203]. The proposed model states grid cells in multiple scales form a scale-space representation of transitions. Scale-space theory itself is well-known, especially in the computer vision community [213]. However, it has not been applied to the concept of transitions and neural spatial navigation previously in the form it was used in this thesis. Influential other or related work is clearly marked at the appropriate places.

At the time of writing, the results presented in Part II have not been reported in a peer-reviewed publication yet. However, valuable feedback was collected during and after a presentation of the matter at Ludwig-Maximilians-Universität München on 14th of February 2017, hosted by Andreas Herz. Furthermore, a preprint that outlines the results presented in Part II is available as

N. Waniek. Multi-Transition Systems: A theory for spatial navigation.

The manuscript uses several parts of the thesis verbatim due to the technicality of the content. For instance, theorems and proofs are taken as-is. In addition, several figures are reproduced.

The algorithms and data structures presented in Part III, Chapter 8, were developed in collaboration with Edvarts Berzs. They were conceived and evaluated during his Master's thesis [22]. Several figures of the chapter were reprinted from his thesis with permission. Furthermore, the pseudo-code for the algorithms given in Appendix D and the complexity analysis which was derived collaboratively and reprinted in Appendix E, are taken as-is, also with permission. The results were submitted for peer-review as

N. Waniek, E. Berzs, and J. Conradt. Data structures for locally distributed routing.

Figures that are displayed in this thesis and reprinted or adapted from others, for instance from the Master's thesis [22] or the submitted manuscript [372], are clearly marked as such. Any other figure is the work of the author.

### 1.4.1 List of Publications

The following list contains publications that were accepted at the time of writing and submitted or prepared during the phase of the dissertation. In addition, submitted but pending publications and manuscripts still in preparation are listed.

#### Accepted peer-reviewed journal papers

1. M. Mulas, N. Waniek, and J. Conradt. Hebbian plasticity realigns grid cell activity with external sensory cues in continuous attractor models. *Front Comput Neurosci*, 10:13, Feb 2016.

#### Accepted peer-reviewed conference papers

1. N. Waniek, J. Biedermann, and J. Conradt. Cooperative SLAM on small mobile robots. In *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1810–1815, Dec 2015.
2. N. Waniek, S. Bremer, and J. Conradt. Real-time anomaly detection with a growing neural gas. In *Artificial Neural Networks and Machine Learning – ICANN 2014, volume 8681 of Lecture Notes in Computer Science*, pages 97–104. Springer International Publishing, 2014.
3. R. Araújo, N. Waniek, and J. Conradt. Development of a dynamically extendable spinnaker chip computing module. In *Artificial Neural Networks and Machine Learning – ICANN 2014, volume 8681 of Lecture Notes in Computer Science*, pages 821–828. Springer International Publishing, 2014.

#### Accepted conference and workshop posters

1. N. Waniek, J. von Stetten, and J. Conradt. Event-based graph cuts, 2016. Poster presented at *Neurocomputing Systems Workshop, Frauenwörth, 2016*.

2. M. Mulas, N. Waniek, and J. Conradt. Exploiting grid cell properties for robotic spatial navigation. Poster presented at *BCCN Retreat, Tutzing, 2015*.
3. N. Waniek, M. Mulas, and J. Conradt. Self-organization of grid cell networks. Poster presented at *Bernstein Conference on Computational Neuroscience, Heidelberg, 2015*.
4. M. Mulas, N. Waniek, and J. Conradt. Neuromorphic architecture for robotic spatial navigation. Poster presented at *Bernstein Conference on Computational Neuroscience, Göttingen, 2014*.
5. N. Waniek, M. Mulas, and J. Conradt. Grid cell realignment based on idiothetic head direction cues. Poster presented at *Bernstein Conference on Computational Neuroscience, Göttingen, 2014*.
6. N. Waniek, C. Denk, and J. Conradt. GRIDMAP – from brains to technical implementations. Poster presented at *Bernstein Conference on Computational Neuroscience, Tübingen, 2013*.
7. N. Waniek and J. Conradt. From brains to technical implementations, 2013. Poster presented at *BCCN Sparks Workshop, Tutzing, 2013*.

#### **Submitted publications and manuscripts in preparation**

1. N. Waniek, E. Berzs, and J. Conradt. Data structures for locally distributed routing. *submitted*.
2. N. Waniek, J. von Stetten, and J. Conradt. Graph cuts for asynchronous event-based vision sensors. *submitted*.
3. N. Waniek. Multi-Transition Systems: A theory for spatial navigation. *in preparation*.

# Chapter 2

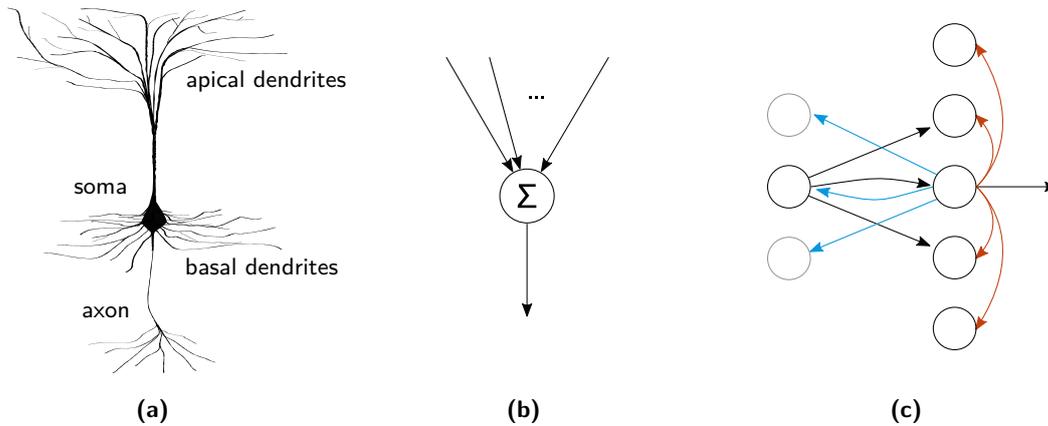
## Biological and artificial neural networks

In the mammalian brain, information is processed primarily by neurons. This chapter thus gives a compact but general overview of neural networks, how to model them, issues evoked thereby, and how it links back to the work presented in the thesis. Furthermore it will point out important influences to the modelling efforts presented in later parts. Readers familiar with the matter may want to jump directly to the end of the chapter, i.e. Section 2.5, where the modelling approach that is used in the thesis is motivated.

The scope of the research area of neural networks in neuroscience and computer science is tremendous. Hence, the cited literature is certainly not comprehensive, but only a curated collection of prominent publications to allow further study on each subject or pointers to examples.

### 2.1 Neurons and synapses

Most biological neurons consist of a dendritic tree to read out activity from other pre-synaptic neurons, a soma, and an axonal tree to propagate activity to post-synaptic neurons [177]. A hand-drawn illustration of a pyramidal neuron is shown in Figure 2.1a. At rest, a neuron's membrane potential is actively maintained at around  $-65\text{mV}$  with the help of several different types of ion channels [186]. When a neuron receives excitatory input, its electrical membrane potential changes from a resting potential towards a certain threshold [121, 148, 177, 186]. Given enough input, this threshold is reached at which point several ion channels in the membrane open abruptly, thereby inverting the membrane potential. This sudden change is called spike and, due to electrophysiological properties of the membrane and its contained ion channels, travels along the dendritic tree towards the soma [148, 186]. On the soma, activity from different branches of the dendritic tree are integrated. In case of sufficient collective input to allow the membrane potential to reach the spiking threshold on the soma or, more detailed, the axon hillock, a spike is propagated along the axonal tree to post-synaptic neurons. Hence, a neuron collects activity from pre-synaptic neurons and, given sufficient input, propagates information to post-synaptic neurons [177]. Neural responses over time are usually called spike trains and are subject to variabilities [121]. The spike response of a single neuron depends not only on the input from pre-synaptic neurons, but additionally on the internal state of the neuron. For instance, in a short time window after a spike, which is called absolute refractory period, a neuron cannot reach the spiking threshold. This



**Figure 2.1 – Hand-drawing of a neuron, abstraction, and different types of connectivity.** (a) The hand-drawing of a pyramidal neuron shows rich branching of both apical and basal dendrites. Information in form of pre-synaptic spikes is collected by the dendrites and forwarded to the soma. A neuron’s axon projects spike activity forwards to other neurons. The arrows indicate direction of spike propagation. (b) Neurons are commonly simplified in computational models, e.g. to a point-model representation which integrates pre-synaptic activity (indicated by the symbol  $\Sigma$ ) without simulating details of spike-propagation on the membrane of the neuron. (c) Several neurons form networks in which they are connected via feed-forward projects (black arrow), recurrent connectivity from one layer back to previous layers (blue arrows), or collateral recurrent connectivity within one layer (red arrows).

period is followed by a relative refractory period in which the membrane potential is below the resting potential, thus making it difficult though feasible to push the neuron to a spiking behavior [121, 186]. Other variables, many of which are still subject to ongoing research, may have an influence on the spiking dynamics of a single neuron, for instance certain neuro-modulators. In case an elaborate and in depth discussion on modelling the electrophysiological properties of spiking neurons, spike propagation, and ion channels is required, the reader is kindly referred to [186].

The input region to a (sensory) neuron is called *receptive field* [177]. Usually, a sensory neuron responds to only one or a few stimuli from its input space, but not to others. Neurons of the same module, i.e. neurons which express the same functionality and are co-localized, usually have overlapping receptive fields. Thereby they densely sample the input space, which can lead to optimal representations of continuous input variables given only finite and discrete numbers of neurons [80]. The specific, often bell-shaped, form in which a sensory neuron responds to input is called tuning curve and is often modelled as a Gaussian function or a von Mises distribution, centered on the input to which a neuron responds maximally [80]. Neurons in higher cortical areas of the brain express more complex tuning curves and receptive fields, however most of which are not or only rudimentarily understood. Several approaches for modelling sensory input and neural responses, neural representations, and learning aspects are described in the literature [80]. One particular form of modelling neural inputs and neural activity employs probability theory. Here, neural activity and receptive fields are described in terms of Probability Density Functions (pdfs) which

allows to treat them with mathematical tools, for instance Bayes theorem [80]. It is also possible to derive optimality constraints for probabilistic population codes, i.e. the coding scheme for multiple neurons participating in a probabilistic computation, and understand post-synaptic neurons as decoders for the probabilistic population codes. In addition to Dayan et al. [80], a general overview of Bayesian approaches to modelling neural activities and encodings can be found in Doya et al. [88].

Individual neurons and networks thereof can be modelled with varying levels of granularity [62]. Multi-compartment models seek to simulate small patches of a neural membrane tissue to be as physiologically plausible as possible, sometimes even with accurate dynamics for individual ion channels [186]. The other end of the spectrum is network models which operate on representations of the whole network dynamics instead of individual neurons [79, 88]. Many models rely on representations that lie in between these extrema, e.g. single compartment models in which the potentially non-linear dynamics of a whole neuron are reduced to only a few equations [32, 163]. A point-like neuron which integrates several inputs is illustrated in Figure 2.1b. Certainly the type of model needs to reflect the purpose of the modelling effort and the question that should be answered. For instance, associative or feed-forward networks which express content-addressability can be modelled as simple neurons with binary synapses [151, 270]. In many cases this is sufficient to understand network properties, distributed computational principles and parallel processing capabilities, or to describe certain effects of neurons in real biological networks [181]. Furthermore, large networks of single-compartment neurons can easily be constructed and simulated in real-time on commodity hardware [163]. On the other hand though, modelling the emergence of certain properties of a neuron may rely on non-linear temporal dynamics for which an increased level of detail is necessary [21, 111, 222]. Without dedicated hardware such as specialized neuromorphic chips or huge amounts of conventional computational resources it is almost impossible to simulate detailed models in real-time, rendering the numerical analysis of the network dynamics a time-consuming process especially in the case of large quantities of neurons. In such complex models, a mathematical analysis is often impossible due to inherent non-linearities. Thus, the complexity of neurons and network models is usually reduced as soon as possible not only to lower the computational workload but also to make analytical treatment feasible. For single-neuron dynamics, important contributions with respect to simplifications, computational efforts, and analytical treatments were proposed by Fitzhugh, Brette, or Izhikevich [32, 109, 163]. Gerstner [121] and Izhikevich [162] present general introductions to modelling dynamical systems for spiking neurons.

## 2.2 Neural networks and associative memories

Biological and artificial neural networks are often massively interconnected [121, 144, 177]. Important to note is that spike propagation from one neuron to another does not happen instantaneously in real biological networks [186]. Not only passes time while the spike is propagated from one neuron to another, but also the propagation along a neuron's axon or dendritic tree consumes time. As a consequence, temporal dynamics are induced by these short latencies which can lead to certain but important properties within the connectivity structure of the networks. For instance, strengthening or

weakening of connections between two neurons, or the self-organization of a whole network, are affected by latencies [121, 393].

The connectivity between neurons can be classified according to their directionality and targets [144]. The best studied form is feed-forward connectivity, in which one layer of neurons acts as input to another layer of neurons. Furthermore, neurons that are connected to their close physical neighbors which belong to the same layer or functional group of a processing stage are said to have collateral connections [177]. On the other hand, recurrent connectivity covers both long-range collateral connectivity within one layer as well as connectivity across layers in such a way that the information flow through the system forms a loop. The three forms of connectivity are illustrated in Figure 2.1c, which shows two layers of simplified neurons connected in the described ways. It is generally assumed that recurrent connectivity is essential to maintain state over longer periods of time [5]. The resulting network dynamics express non-linear behavior and are therefore difficult to examine theoretically. Nevertheless, theoretical assessments of the dynamics with the help of non-linear system theory were very successful in describing the state-evolution of neural networks, e.g. [311]. In many cases though, numerical simulations are the only currently available tool for examination.

Recurrent connectivity can be used to model higher cortical functions in form of Continuous Attractor Neural Networks (CAN) [5, 151]. The activity within such networks tends to converge to a certain state, the network's attractor, which can be kept active over longer periods of time due to recurrent and collateral connectivity. The shape of the attractor may vary, e.g. it may be a single point of activity within the network or be expressed in form of lines or other, more complex shapes. For instance, in neural networks with overlapping receptive fields and recurrent connectivity, a point attractor may emerge which is expressed as a single bump of neural activity. The computational principles of such continuous attractor networks and how they could be implemented in real neural networks were recently described in depth in [54]. CANs are also thought to form the basis of associative memories.

Neural associative memories can be used to store and retrieve patterns and are able to maintain their activity over longer periods of time [4, 151, 270, 273]. In the case of an auto-associative memory, patterns  $u_0, \dots, u_M$  are stored during the learning phase. Afterwards, the patterns can be retrieved during the retrieval phase by addressing the content of the memory even with input which is distorted by noise. Therefore, an auto-associative memory performs pattern-completion. In the case of a hetero-associative memory, input patterns  $u_0, \dots, u_M$  are used to store and associate with output patterns  $v_0, \dots, v_M$ . Here, the memory will return an output pattern  $v_i$  during the retrieval phase when addressed with a potentially noisy input pattern  $u_i$ . In neural associative memories, the response often differs from the optimal solution or stored pattern, measured in terms of the retrieval [182]. The attractor in CANs may vary due to the internal re-configuration of the network, or may change over time due to external input to the network [80, 151, 270]. The way in which the state changes typically depends on the form of the recurrent connectivity as well as other influences like the temporal dynamics of inhibitory inter-neurons. In most network models, the recurrent connectivity is pre-defined or learned previously to employment of the network. Then, an unspecific trigger signal is sufficient to toggle transitions and perform syntactic sequencing of consecutive neural activity states in

artificial neural networks [377]. Sequences of assemblies of neurons were observed and described for activity in the rodent Hippocampus [279]. This area is believed to consist of associative memories [299], and is of significant interest in Part II of this thesis.

The currently active neural state can be considered to be distributed [270]. The co-activity of neurons in several distributed modules forms what is called a neural or Hebbian cell assembly [46, 273]. Assembly theory was successfully used in several models to explain the distributed activity of the brain and how it could emerge naturally from associative memories [212, 286]. Furthermore, the theoretical treatment demonstrated the capabilities to parallelize and distribute computations [29, 376, 377].

The results of the theoretical and modelling efforts of the last few decades allow to interpret some neural networks, and associative memories in particular, as distributed content-addressable memory [46, 273]. In common hardware and if the input patterns can be matched exactly, this form of memory can be implemented efficiently using tables and hash-functions for table-lookups [182] or, generally speaking, hash-tables. If the receptive fields of a neural network are topographically arranged, the hash function can be furthermore considered locality-preserving, which allows improved parallelization [60]. However, neural associative memories are advantageous when the input patterns cannot be matched exactly. For instance, noisy inputs in which only partial patterns are available can be reconstructed in neural networks but will pose issues in associative memories with common hash functions [151, 270]. Furthermore, the usage of overlapping receptive fields provides a means of generalization over the input space which is difficult to achieve in hash-table implementations without additional effort [355].

In many models using associative memories, a single neuron associates only with one specific input pattern. However, the complexity of dendritic trees of real neurons likely allows association with multiple patterns. Recent studies indeed demonstrated learning on individual branches of the dendritic trees [30, 365]. It is therefore likely that neurons provide a mechanism which allows multiple entries to be stored by one single neuron.

## 2.3 Plasticity, synchronization, and learning

A fundamental property of neural networks is plasticity [177]. The strength of connections between biological neurons is usually not pre-defined but the result of an ongoing learning process. In models which use rate-based neurons, i.e. a representation of neural activity in which only a neuron's firing rate over time is considered but not individual spikes, typically one of several forms of Hebbian learning is employed, named after Donald Hebb who initially proposed the mechanism. Here, the strength is the result of the correlated activity between neurons [145, 252].

One specific learning rule which is biologically plausible in the sense that it allowed accurate predictions for learning in the visual cortex is the Bienenstock-Cooper-Munro (BCM) learning rule [25]. However, it was long unclear how this form of learning could happen on the level of spiking biological neurons until it was discovered that precise spike timing of pre- and post-synaptic neurons is important [81]. In simple terms, the connectivity between two neurons is strengthened if the pre-synaptic neuron spikes just immediately before the post-synaptic neuron and depresses if the

opposite sequence of events occurs. This finding gave rise to what are now called STDP learning rules [23, 78, 393]. It was demonstrated that the BCM rule and STDP learning rules are related [163]. However, the precise mechanisms underlying or the form of STDP are still not fully agreed upon. Evidence suggests a certain spike-triplet rule which was, for instance, found in the visual cortex [283]. This rule requires a fast rate adaptation of neurons to provide stable results during longer time scales, increasing the complexity of the models [393]. The rule can be altered further to allow rate-based or even one-shot learning [144, 182, 270, 312].

The synaptic efficacy can be modified in several ways. On the one hand, there exists Long-Term Plasticity (LTP) which modifies the synaptic strength sustainably such that the weight change persists even after longer periods of time [177]. On the other hand, Short-Term Plasticity (STP) induces changes in the synaptic efficacy which last only short time windows after which the strength converges back to its previous state, and has been observed for neurons in the Hippocampus [308]. LTP is assumed to be the result of repeated stimulation of pre- and post-synaptic neurons and the ensuing growth of novel or strengthening of existing dendritic spines, and postulated by Donald Hebb in 1949 [145]. In the case of STP, the mechanism is not as well understood. In both cases however, evidence suggests that initial association happens comparably fast after presentation of only few input patterns. In the extreme case this could be considered one-shot learning, i.e. learning of an association after the presentation of only a singular input pattern. This kind of learning is typically used in associative memories [180, 181], but was only recently employed successfully in other artificial neural networks [312].

Despite the asynchronous nature of neurons, synchronization of neural activity was proposed to solve the binding problem and is often required for fast learning processes [298, 376]. Distinct sensory modalities are extracted by different neurons, for instance neurons encoding for the orientation of a stimulus or neurons which encode the color [130]. However, it is necessary to bind the representations in such a way that a coherent internal perception of contiguous external stimuli emerges [298]. For example, the neural representations emerging for a red apple may differ from the activity for a green fruit knife. Still, both objects may be perceived at the same time. Due to findings primarily in the cat visual cortex, the temporal correlation hypothesis was put forth which states that the binding problem is solved by synchronization [92, 369, 370]. Summarized, neurons which fire in response to the same stimulus will correlate their activity which will, in turn, lead to synchronized firing of neurons corresponding to a singular stimulus. The superposition of different stimuli is thus resolved by a temporal coding scheme. It is thought that the result of synchronized behavior of neurons can be observed in Electroencephalogram (EEG) as certain oscillatory waves. However, it is not possible to derive single neuron behavior from EEG signals. While EEG data is recorded on the surface of the scalp, Local Field Potentials (LFPs) are retrieved from the electrical potential in the extra-cellular space within brain tissue. Several brain waves were identified both in EEG and LFP which are linked to behavior [177]. One notable wave is the Theta frequency, which oscillates at 4 to 10 Hz [48]. Initially believed to be the result of loops of activity within neural networks or effects of summing membrane potentials, it was later proposed that the observed oscillations are due to synchronized activity of neurons [91]. Currently, synchronized synaptic currents are held accountable for the reported measurements.

Regardless of the detailed mechanism providing the necessary plastic substrate, there are three major strategies of learning considered in the literature. During supervised learning, a network of neurons is presented with an input signal and computes an output. The output signal is compared to an ideal training example. Given the computed output and the training signal, an error can be computed which in turn is used to adapt the synaptic weights within the network, for instance using the well-known error back-propagation algorithm [304]. On the other hand, there is no immediate error signal during unsupervised learning due to the lack of a training signal. Here, the neural network generates internal representations of the input data, for instance by forming a self-organized map due to local neighborhood descriptions or recurrent connectivity [51, 191]. Self-organizing maps usually operate on a fixed number of neurons and static descriptions of the neighborhood and synaptic connectivity of neurons. An extension named GNG allows the generation of novel neurons within the system and to adaptively change the neighborhood of each neuron [113]. Finally, Reinforcement Learning (RL) can be considered to fall somewhere in between supervised and unsupervised learning [340]. Although an RL system receives a training signal in form of a reward, this reward may only indirectly relate to system's learning objective and thus can be considered non-supervised. Usually, an RL system learns to generate a sequence of actions which is optimal with respect to the given reward. For instance, an agent that needs to travel from one location to another may find one sequence of actions more rewarding than another. For this purpose, an internal representation of state-action pairs can be self-organized which helps to solve an optimal control problem to maximize the reward. Thus, RL may fall into the area of unsupervised learning. However, most RL systems exploit knowledge about the world and thereby the state-action pairs to predefine necessary structures to hold the information about future rewards, moving the systems more towards classical supervised learning strategies. Nevertheless, evidence suggests that some areas of the brain in fact operate on principles of RL [79, 223].

## 2.4 Modularity and hierarchical computation

Inter-connected areas in the brain form small-world networks [176]. Modules are connected only to few others instead of exposing all-to-all connectivity. Usually, each area of the brain has a specific computational purpose [177]. For instance, motion perception or landmark detection are computed in one or multiple distinct areas. The distributed co-activation of neurons within several modules form a distributed neuronal assembly. Regional connectivity structures forming local modules are well studied and classified according to their cytoarchitecture or organization following Brodmann [33]. Modules are often distinguishable from surrounding areas due to the cellular structure and morphology of the region, and just recently novel areas were identified [123]. On the basis of lesion studies, many of the regions are well characterized in terms of their general computational purpose.

Findings suggest that hierarchical and multi-layer computation is a successful strategy to increase the complexity of computed functions. For instance, hierarchical models exposing distributed asynchronous computation with spiking neural networks were used to model the auditory cortex as associative memories [29]. However, interactions between modules and hierarchical computations are not as well understood

as regional connectivity. Recent advances improved the current understanding of the connectivity within and across several brain areas and of hierarchical representations [261]. Nevertheless, there are many unresolved issues. The challenge in forming hierarchical computational models for neuroscience is the necessity to understand the functional properties of the computation as well as the connectivity structure within communities of connected modules [230]. Knowledge thereof is frequently limited, as single neurons may react differently depending on context. For instance, they may express increased activity on the detection of novel signals, but remain silent in another context. This issue is currently addressed from several directions, e.g. information theory [2, 3, 272]. In other models, Bayesian inference is used to describe not only the neural representation within single modules but also to describe the computation across hierarchies [208]. Although it is still debated whether the brain performs Bayesian inference, probabilistic models were able to accurately predict neural response patterns [80].

## 2.5 On the employed modelling approach

Modules and interactions between areas can be represented in graph theoretical notations [37, 275, 276, 333]. This makes it feasible to not only apply methods and techniques from graph theory to analyse network dynamics or build hierarchical models [208] (for a recent survey of Bayesian networks applied to neuroscience, see [24]). Given sufficient data, it also allows to determine structures and interactions within the cortex [261]. The challenge is thus to record from multiple sites at the same time and establish a proper theory for the function computed in each area. However, recording from real neurons is difficult. This is especially true for behavioral tasks in which an animal needs to move freely such as spatial navigation. The instruments and computers required for recording are mostly stationary and only recently novel techniques were developed to address this issue [139]. Furthermore simultaneous recordings of neurons are often limited to only tens of neurons at once except for very few remarkable studies, e.g. [282],

Regardless, it seems appealing to model neurons in spike-based networks. Some properties and emerging effects may only be due to the temporal dynamics of spike propagation. For instance, specific observations in the visual cortex can be explained primarily by examination of precise spike timings [347, 363]. Although spike timing has been successfully used to learn visual features with an unsupervised learning strategy based on STDP [232], the non-linear dynamics often make mathematical analysis infeasible. Furthermore, it is arguable if the details and parameter tuning required for spiking models is necessary to understand the computational functionality and purpose of a neural network.

Conclusively, abstractions of the network dynamics are introduced as early as possible in this thesis. For instance, the feasibility of certain assumptions is demonstrated in biologically plausible spiking neural networks. Afterwards, the results are simplified to allow implementation in larger systems or mathematical treatment. By using a mixture of top-down and bottom-up approaches, the focus is thereby set on computational functionalities and less on dynamics of individual neurons.

## Chapter 3

# The neural representation of space

Mobile agents need to sense their environment to interact with it [244]. To navigate towards a goal, perception of the goal location or some form of a gradient which leads towards that goal is necessary. However, perception of a goal is not always given. Hence, explorative movement should build up memorizable knowledge, which can be used later to retrieve a trajectory even in the absence of goal perception. Memorizing and retrieval of spatial knowledge appears to be of behavioral relevance also for animals. Consider an animal roaming for food for its cubs. As soon as the animal found suitable nourishment, it has to trace back a path to its shelter to provide the food to its progeny and not get caught by a predator. Thus, the animal needs to read out sensory organs and internally represent and link their states while it was searching for food. Furthermore, the animal should be able to find shortcuts to closeby locations to avoid unnecessary detours.

In rodents, these navigational, memorizing, and cognitive tasks are computed primarily in sub-areas of the Hippocampus and the para-hippocampal region, including the subiculum and Entorhinal Cortex (EC) [26, 265, 300]. The information flow within and across these areas form several loops of information processing [7, 364]. When disrupted, goal-directed navigation is severely impaired [251]. Several types of neurons participate during the computation, and their characteristics and interactions will be the focus of this chapter.

### 3.1 The Hippocampal Formation and Entorhinal Cortex

Located in the medial temporal lobe (see Figure 3.1 for a simplified illustration), the Hippocampal Formation (HF) is compartmented into several sub-areas [364]. The main sub-areas are considered to be the Dentate Gyrus (DG), Cornu Ammonis 3 (CA3), Cornu Ammonis 1 (CA1), as well as the subiculum, all of which contribute to spatial information processing [7, 364]. The small area between CA3 and CA1, Cornu Ammonis 2 (CA2), is under-represented in current hippocampal research and therefore its contribution to represent spatial information is not properly understood. The Cornu Ammonis sub-fields are commonly called Hippocampus, and together with DG they form the Hippocampus proper [7].

Located adjacent to the Hippocampus lies the EC. Containing several spatially modulated neurons, it is considered one of the major inputs projecting to the Hippocampus and was one of the major areas of research during the last few years with respect to spatial information processing in the brain. It is subdivided into

### 3. The neural representation of space

a medial and a lateral part, the Medial Entorhinal Cortex (mEC) and Lateral Entorhinal Cortex (lEC), respectively [7, 364].

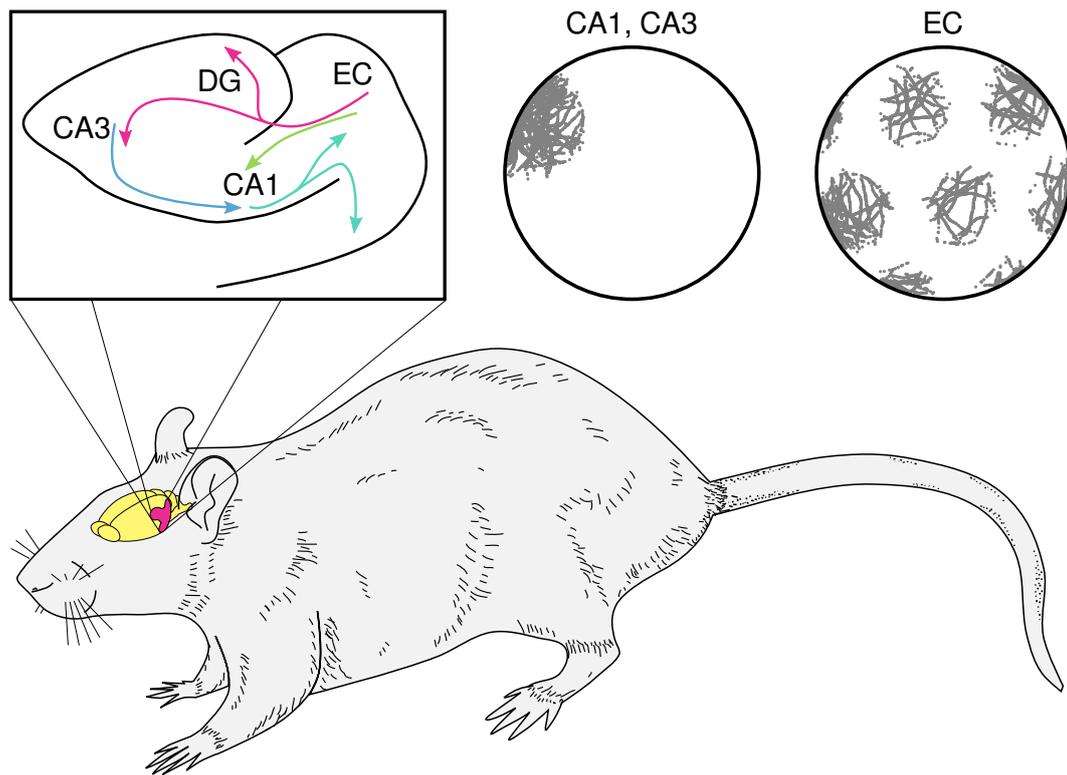
There are currently two major lines of thought considering the general functionality and purpose of the Hippocampus. On the one hand, it is attributed to episodic memory [229, 319, 358], i.e. the storage and retrieval of sequences of behaviorally relevant information [58, 107, 167, 364]. On the other hand, behavioral studies demonstrate its relation to space [125, 265, 383]. Here, it is perceived as the fundamental substrate for the biological equivalent of a navigational system [76, 249, 251], for mismatch correction of spatial knowledge [126], or generally speaking navigation and spatial memory consolidation [12, 167]. Important for spatial navigation, the network expresses one-shot learning capabilities [260].

Findings suggest that the DG plays a crucial role in mapping compressed inputs from the EC to a high-dimensional space. Thereby it decorrelates representations and performs pattern separation [210, 229, 239]. The objectives computed in CA1 and CA3 are primarily related to spatial and episodic memory [300, 357]. The precise functionality of most other sub-areas of the Hippocampus remain elusive. They express complexity both in organization as well as interaction and are attributed to declarative memory formation [364].

The inter-area connectivity including the Hippocampus and EC is considered to form a poly-synaptic circuit called *trisynaptic loop* [6] (see information-flow inlay in Figure 3.1). Local recurrent connections within the contained areas indicate additional nested loops of information processing [11]. The para-hippocampal region is organized and structured in six distinct layers, sharing similarities with the neocortex [50, 364, 386]. In contrast, the HF is organized in only three layers, namely the polymorphic or deep layer, a central layer, and a superficial layer. The perforant path is a unidirectional projection from EC to the Hippocampus, primarily to sub-areas DG, CA3, as well as CA1. Although the axons to CA3 origin mostly in layer II of the EC, several projections from layers III, V, and VI exist [386]. In contrast, the projections to CA1 origin mostly in layer II [364]. In the Hippocampus, the information flows mostly sequentially on unidirectional projections via mossy fibers from DG to CA3, and by Schaffer collaterals projecting onwards from CA3 to CA1 [107, 190]. In reverse, the Hippocampus back-projects to the EC.

Local recurrent connectivity suggests that CA3 forms an auto-associative memory [205, 241, 245, 277], likely in form of a CAN [356]. In contrast, CA1 presumably forms a hetero-associative memory [299, 386]. In addition to indirect excitatory recurrences, several direct recurrent connections or indirect couplings via inhibitory-interneurons were discovered in the Hippocampus and EC [115]. Curiously, the recurrent connectivity within mEC was found to be predominantly inhibitory [70].

The sub-areas of the Hippocampus are interconnected with multiple other areas in the rodent brain [7]. For instance, connections from and to the Pre-Frontal Cortex (PFC) exist and are necessary for spatial navigation [161, 284]. In turn, PFC is attributed to decision making and involved in the formation of long-term memories [102, 192, 284]. This suggests that PFC has the capability to govern the activity in the Hippocampus, e.g. by suppressing or facilitating specific neuronal responses which are related to the animal's current desire [124]. Furthermore, connections towards and back-projections from the subiculum were reported [107]. The recurrent connectivity of hippocampal areas is considered to be essential for goal-directed navigation [179].



**Figure 3.1 – Schematic of areas in the rodent brain and exemplary spike responses of place and grid cells.** The main areas of interest in the rodent brain (yellow area) for spatial navigation are the Entorhinal Cortex, and Cornu Ammonis 1 and 3 (EC, CA1, and CA3; all highlighted in magenta). The information flows from the EC across the Dentate Gyrus (DG) to the CA3 and CA1 regions and recurrently back to the EC, forming the *trisynaptic loop*. Single neuron recordings of pyramidal neurons from CA1 and CA3 mostly express a singular area of activity with respect to the location of the rat and are thus called *place cells*. The inlays show examples of typical cell responses in simulations in a circular environment for either CA1 and CA3, or EC. In contrast to place cells, stellate cells of the EC respond in regularly arranged locations and are called *grid cells*. In both examples, each gray dot represents a spike of a single neuron with respect to a circular arena.

### 3. The neural representation of space

LFPs recorded in the Hippocampus revealed an oscillation which is highly regular at a frequency of around 4 – 10 Hz while the animal is moving [48]. While the reason for this oscillation, termed Theta, is commonly attributed to network mechanisms, its true origin and purpose are not conclusively agreed upon. Observations suggest that the oscillation coordinates interactions between the Hippocampus, PFC, and other extra-hippocampal areas, thereby supporting decision processes and memory consolidation [140, 173]. The oscillation was also perceived to be important for short-term memory [366]. Furthermore, Theta is considered to be the result of neural activity traveling in form of waves and therefore a synchronization mechanism within the Hippocampus [221]. Theta oscillations were also discovered in the EC and thus linked to spatial memory formation [47], where it is considered to buffer temporal information, and separate retrieval and encoding of memories [140].

Neurons in the Hippocampus and the EC show activity which is temporally relative to Theta, an observation called Theta phase precession [169, 328]. When an animal is moving along a trajectory, the currently best matching place cell is spiking at the trough of Theta, while the place cells which correspond to places before (after) the current location are active on the upward (downward) slope of the oscillation. Theta and Theta phase precession are illustrated in Figure 3.2. Likewise the Theta rhythm, this salient behavior is only partially understood and probably due to network mechanisms or other intrinsic dynamics [235]. Nevertheless, one of its main purposes is believed to form a compressed representation of temporal information [328]. Theta phase precession can thus be interpreted as the observable operation of a temporal buffer structure [188, 246]. Recently a link between Theta phase precession, spatial information, and reward modulation was suggested [362]. Interestingly, Theta phase precession of the Hippocampus was reported to be independent from Theta phase precession of the EC [134], indicating that it is either a general purpose mechanism or effect due to network dynamics, or corresponds to synchronization properties of afferent inputs [48, 173].

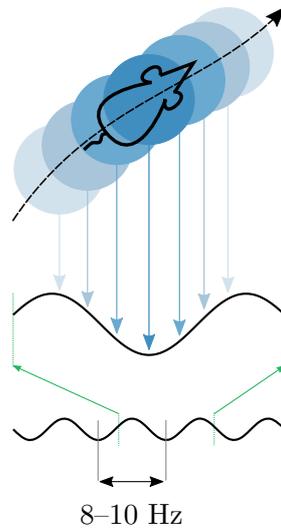
To summarize, the Hippocampus and EC are considered two major areas responsible for episodic memory, spatial navigation, and spatial information processing. The flow of neural activity within and across the two areas forms several, potentially nested but distributed and concurrently operating, loops of information processing.

## 3.2 A zoo of spatially modulated neurons

The Hippocampus and EC contain several spatially modulated cells. Data suggest that the neurons living in the EC form encoders of rather specific sensory and spatial modalities, whereas neurons in the Hippocampus form more generalized or abstract representations [131, 386]. The following overview will present the cell types which are the most important for the work presented in the thesis, and addresses some mechanisms which are believed to be the origins for the neural responses.

### 3.2.1 Place cells

Place cells are pyramidal neurons initially discovered in the Hippocampus in 1971 [264, 265]. A place cell typically expresses activity only when the animal is in a singular area within an environment, the cell's *place field* (see inlay *CA1, CA3* in



**Figure 3.2 – Theta and Theta phase precession.** Theta is an oscillation observable in Local Field Potentials (LFP) of neurons in the Hippocampal Formation (HF) at around 4 – 10 Hz (black sine wave at the bottom). The spike times of neurons encoding spatial locations are relative to Theta when the animal is moving (black dashed arrow). The cell corresponding to the animal's current location spikes at the trough of Theta. Cells encoding past and future locations are active relative to the trough. Cells and their corresponding *place fields* are indicated as circles, their relative spike time as arrows pointing to a zoomed-in area of Theta (partial sine wave in the middle).

Figure 3.1 and Figure 1.1a), or only very few small areas but without a perceivable regularity in the firing field locations [278]. Especially in large open environments, data suggests that place cells tend to generate multiple but irregularly spaced place fields [105]. Nevertheless and once formed, the representation encoded in place cells is mostly stable even after weeks [346], i.e. a place cell represents the same location across recording sessions. Studies showed that the sizes of place fields depend on experience and increase from 20 cm to about 35 cm on average [243]. Another study pointed out that place field sizes additionally depend on the type of trajectory [380]. Here, average field sizes between 35 cm and 45 cm were reported.

Both, areas CA1 and CA3, express place cell activity, but the functionality was reported to be heterogeneous and not uniquely dedicated to localization [207]. Several models were suggested to explain or describe the firing characteristics of place cells [13, 321, 357]. In these models, place cell firing is usually driven by other spatially modulated neurons, e.g. boundary cells which will be discussed later. Other data suggest that place cell firing corresponds to goal-directed trajectories [380].

Evidence suggests that spatial representations in the Hippocampus form due to the influence of other spatially modulated neurons [259]. However, stable formation of place fields after removal of distal cues or visual sensory information, e.g. during experiments in total darkness or with blind rats, suggests that place cells rely not only on afferents carrying visual information but also on other spatially modulated cues [314, 394].

Later studies reported place cells or neurons with a confined spatial correlate outside the Hippocampus [287, 315]. However, it is unclear whether the reported

cells are indeed place cells, recording artefacts of other cell types, or a completely separate type of cell.

Place cells perform remapping on detection of novel environments, a process which is linked to the formation of memory [67]. On a remapping event, the place fields of place cells appear to be re-located to arbitrary other positions. Furthermore, the relations between place fields of different place cells are not preserved. For instance, two place cells which expressed place cell firing fields which were previously co-located can exhibit firing fields which are separated by larger distances (see Figure 3.5a for an illustration). However, a previously encountered environment will induce the same place fields on re-entry even after longer periods of time [346]. In addition to full remapping of the place cell firing fields, place cells express another but subtler form of remapping called rate remapping [332]. Here, the firing rates of place cells change but the place fields are (partially) kept. It is assumed that rate remapping allows super-position of spatial as well as non-spatial information at the same time within one network [89].

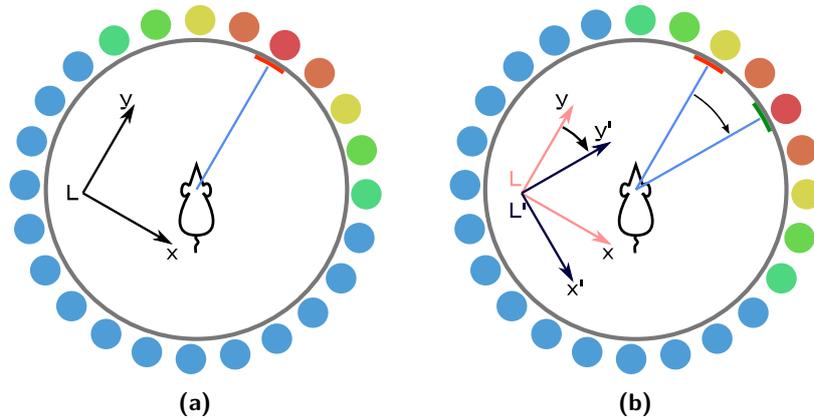
Place cells express activity during trajectory planning and memory consolidation. The latter appears to occur during short bursts of highly synchronization activity, known as *Sharp Waves and Ripples (SWP-R)* which were reported both in sleeping and awake animals [45,72,164,224]. "[The activity arises] from the excitatory recurrent system of the CA3 region and the SPW-induced excitation brings about a fast network oscillation (ripple) in CA1" [45]. Coordinated by several different types of interneurons, sequences of neuronal activity that were experienced previously are replayed during SWP-Rs, potentially strengthening synaptic connectivities and thus consolidating memories [164]. It was observed that SWP-Rs have an impact not only in the Hippocampus but throughout the entire cortex [288], likely supporting spatial memory consolidation in extra-hippocampal regions [164].

Data indicates that path planning processes executed in the rodent brain compute *forward-trajectories* to target locations [282]. In experiments with freely roaming animals, place cell activity was found to compute trajectories to their home locations, outbound from the current location of an animal, in short bursts of activity. In addition, multiple possible trajectories were computed before a winning trajectory was selected. The neural basis for the selection mechanism which favors one trajectory over another is not described in literature, though. Although not finally determined, it is likely that both processes, memory consolidation and path planning, are due to the same underlying mechanism and computational principle.

### 3.2.2 Head direction cells

The dorsal presubiculum, a small area sending many mono-synaptic afferents to the EC, hosts neurons which respond to the Head Direction (HD) of an animal [56,289]. Later, HD cells were found in several other areas throughout the mammalian brain, which suggests that they are an essential ingredient in the processing of spatial information.

Different HD cells express preferential tuning to individual but overlapping HDs and are anchored to distal cues. On rotation of the cue the internal representation of the HD follows the rotation (see also Figure 3.3). Thus they can be understood as an internal compass which is independent of the animals movement direction [342,343]. The representation, updated with very short temporal latencies [400], is stable even



**Figure 3.3 – Head direction cell firing follows distal cues.** (a) The visualization depicts a ring of head direction cells (round objects around gray circle) with preferred tuning towards a certain direction (location on the circle), relative to a visual cue (red bar). Firing rate is color coded, with blue indicating a silent neuron and red a strongly firing cell. The head direction network provides a coordinate system  $L$  relative to the cue. (b) On rotation of the distal cue, the firing activity of head direction cells follow the rotation of the cue. Thereby, the internal coordinate system rotates in concordance.

when conflicting rotations of multiple distal cues are induced [390]. To account for these findings and based on observations of the anatomical organization [285], head direction systems are often modelled in form of CANs, e.g. [31], which allow the necessary rapid updates as well as the observed stability.

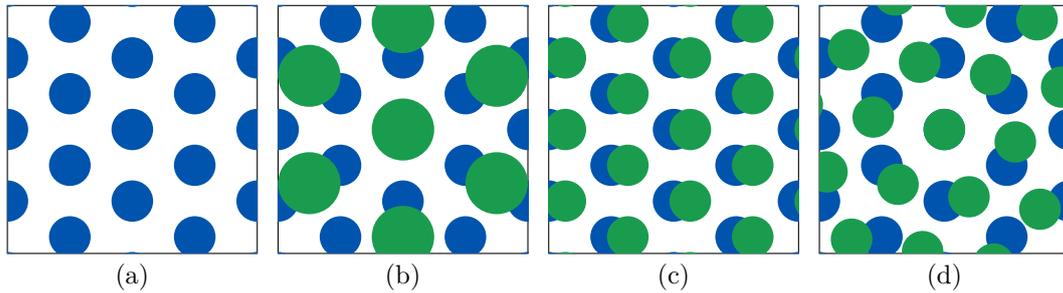
### 3.2.3 Grid cells

Grid cells are stellate neurons with a distinct spatial correlate and were discovered in the rat mEC in 2005 [135]. Later, they were also found in other animals like mice [118], and bats [360, 388]. Although they can be found in all layers of mEC, pure grid cells appear mostly in layer II whereas deeper layers are interspersed by conjunctive cells (discussed further below). Besides place cells, they are considered one of the main contributors to the representational system for spatial information due to their peculiar response pattern [253].

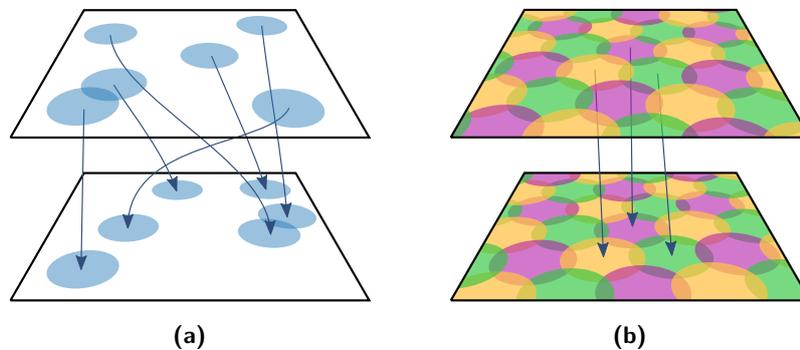
Grid cells express a repetitive pattern of activity, their grid fields, which tessellates an environment into almost perfect triangles (see inlay *EC* in Figure 3.1 and Figure 1.1b). Grid fields are characterized according to the size, phase, and orientation (Figure 3.4), and densely cover the input space due to overlapping fields [256]. The responses are usually evaluated by computing their gridness scores, which determines if and how well the responses form a hexagonal pattern [313] (see also Appendix B). Grid cells appear to be coordinated such that spatial relation between the grid fields of two cells tends to remain stable even across environments [135]. It is thus believed that grid cells generate a metric representation of space [254]. Similar to head direction cells, data suggests that grid cell activity is anchored to external cues as the response follows rotation of distal cues [135]. Hence it is likely that grid cells receive feed-forward input from head direction cells [384].

However, the afferents driving grid cell activity are still disputed. On the one hand, their characteristic response is believed to be the result of path integration,

### 3. The neural representation of space



**Figure 3.4 – Properties of grid cell firing fields.** (a) Idealized firing fields of a grid cell with respect to spatial location in a square environment. The responses of grid cells can be characterized according to size (b), phase (c), and orientation (d) (adapted with permission from [256]).



**Figure 3.5 – Place and grid cell remapping.** (a) On a remapping event from one environment to another, place fields (blue circles) of individual place cells can move to arbitrary locations. (b) Grid cells show remapping in form of a possible change of orientation and shift, however the spatial relationship between grid fields is preserved.

anchored to environmental cues [198]. On the other hand, they require excitatory drive from the HF [28], and depend on the Theta rhythm at least in the rodent brain [317]. Furthermore, indirect evidence suggests that their activity may due to CAN dynamics [391]. However it was reported recently that removal of visual input also leads to a disruption of their firing characteristics, contradicting the hypothesis that the activity of grid cells is self-sustaining or requires only unspecific drive from the Hippocampus [55].

Quite surprisingly, the size and period of grid fields do *not* vary linearly for different grid cells. In contrast, the field sizes cluster around certain discrete scales [256, 336]. Cells of one scale are said to be part of one *grid module*. Remarkably, the scale increment between adjacent grid modules is approximately constant within and even across animals [117]. The increment was reported to be in a range from 1.3 to 1.7, with strong indications that the real value is very close to  $\sqrt{2}$  [15, 336]. This suggests that the brain operates with a form of multi-resolution analysis during spatial navigation, an idea which was successfully included in models and engineered solutions for goal-directed navigation [100, 217]. However, only few currently existing neural models convincingly explain the origin of multiple scales in the light of grid cells [133].

Grid cells express properties related to remapping of place fields [117,248]. On the one hand, grid cell activity rearranges abruptly to a previously learned orientation on re-entry of a known environment but keeping the spatial relationship between grid fields, a process which is termed grid cell realignment. On the other hand, grid fields are influenced by the geometry of the environment [198,315], and follow stretching or shrinking of an environment [228]. Additionally, it was observed that grid cells realign based on other sensory cues such as odor or color of an environment or other visual landmarks [227,280]. Head direction induced rotational realignment based on visual cues was successfully used to model the realignment of grid cells in an existing model of grid cell firing [373]. However, grid fields preserve their neighborhood relations, illustrated in Figure 3.5b. The interactions between hippocampal remapping and Entorhinal realignment are not conclusively understood. Nevertheless, it was suggested that grid cell realignment is the basis for remapping the Hippocampus after observing correlations between the two effects [117,248].

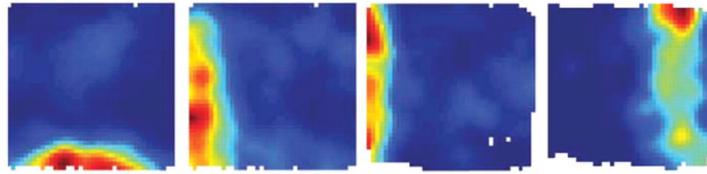
Complex environments induce grid cell firing fields which deviate from an ideal hexagonal firing pattern. In a multi-compartment experiment in which alleys were set up such that they form a zig-zag maze, grid cell firing repeated in every other alley [85]. Other studies showed that boundaries of an environment influence the location of grid fields [197,198]. Notably, grid fields were squeezed and deformed in the apex of triangular environments. Both findings challenge the assumption that grid cells convey almost ideal metric information to place cells.

A recent survey covers the ten years of grid cell research since their discovery [303]. However, a somewhat overlooked finding is that grid cells tend to fire more strongly with respect to head direction and less with heading direction [293]. This observation poses severe problems for models in which grid cells are used as a mechanism for path integration. The movement direction is necessary to properly integrate traveled distances instead of the head direction in these models [122,324,398].

### 3.2.4 Boundary vector cells / border cells

Place cells were found to exhibit firing fields which correlate with the geometry of the environment [266,394]. In the study, place cells expressed elongated place fields along boundaries and walls in one environment, but almost circular firing fields in another environment. It was thus hypothesized that place cells receive input from neurons with spatial modulation depending on geometrical boundaries [13]. The model, developed by Barry et al. [13], suggests that the elongated and deformed place fields could be explained by afferent input to place cells from boundary vector information.

Later studies indeed observed the predicted neurons, responding to geometrical boundaries of an environment [211,315,331]. The cells discharge when the animal is close to a border in a certain preferential direction (Figure 3.6). Called border or boundary cells by different researchers, the cells not only appear in the Hippocampus and mEC, but were recently also discovered in other adjacent areas, for instance the anterior claustrum and the rostral thalamus [165,166]. However it is currently unclear if the boundary cells in these areas share the computational functionality with boundary cells from mEC. Data indicates that they have to be considered a distinct computational group, expressing slightly different responses. According to Grieves et al. [131], boundary cells are "often overlooked" and their interaction with



**Figure 3.6 – Boundary vector cells.** The figure shows four examples of border cells found in the Medial Entorhinal Cortex (mEC). The cells have a preferential tuning towards the distance to and allocentric direction of a geometric border in the environment. The color shows the firing rate with respect to the spatial location of an animal. Red indicates a high, blue a low rate of firing (adapted with permission from [331]).

grid and place cells are complex and ”not greatly understood”.

The cells express tuning not only to a certain distance but also for a specific direction towards the boundary [331]. A boundary cell fires maximally when a geometrical boundary is observed with respect to the neuron’s preferred direction, and not with respect to the animal’s head direction. A suitable afferent for the extraction of boundary information are visual cues, especially optic flow [294].

### 3.2.5 Linear speed cells

Many researchers suggest that grid cells perform path integration due to their repetitive spatial firing fields [39, 116, 280]. Given recurrent connectivity, the activity in the network is presumably translated from one grid cell to another such that the periodic response appears [391]. However, this process requires continuous integration of the animal’s velocity to correctly shift the activity within the network [39]. In 2015, neurons with a firing response that is linear with respect to an animal’s speed and suitable for this purpose were discovered in mEC [195].

The cells are context-free and thus not correlated to any other signal except an animal’s ego-motion. Additionally, the firing rate was observed to be prospective, i.e. the rate is anticipatory of future rather than previous or current running speeds. Summarized, the cells form a functionally independent population amongst the other cells found in mEC [195].

It was not reported by which converging inputs the speed cells are driven. However, previous theoretical studies suggested that ego-motion extracted from optic flow can explain the grid cell response [292, 295]. Hence, optic flow is a likely candidate to provide sufficient information to drive linear speed cells.

### 3.2.6 Conjunctive cells

So far, only neurons with distinct spatial correlates were described. However, the functionality is not always as isolated as presented above. Several conjunctive cells were identified, for instance conjunctive grid cells which express hexagonal grid fields but respond only when the animal is facing in a certain head direction [195]. Furthermore, place cells with conjunctive representations were reported [225, 247].

Theoretically, conjunctive grid cells allow to perform linear look-ahead, ideally suited for trajectory planning and goal-directed navigation [200]. Nevertheless, further research needs to be conducted to conclusively answer if this is really the case in the rodent brain. Furthermore, it is as of yet undetermined if conjunctive grid cells are

the result of combining non-conjunctive cells with other spatial modalities, or if the non-conjunctive cells are the result of integrating their conjunctive counter-parts.

### 3.2.7 Interneurons

The Hippocampus and EC are interspersed with several interneuronal systems [112,174,201]. Although a unique spatial selectivity was not reported for these neurons, they tend to have certain but undefined selectivities with respect to spatial modalities. Furthermore, they play important roles during several spatial computations, for instance during SWP-Rs [45].

Besides excitatory interneurons [114], inhibitory interneurons play an important role during the formation and stabilization of spatial representations [9, 53]. For instance, the inhibitory networks in CA1 and CA3 were reported to express very short latencies, leading to strong synchronization of the network [86]. Furthermore, the recurrent connectivity of grid cells in mEC was found to be governed predominantly by inhibition [36, 70]. Hence, it is likely that inhibitory interneurons are important for pattern separation and gain control of the network dynamics. Given their support during network synchronization, they are also likely to be important during memory consolidation and for solving the binding problem, mentioned in Section 2.3.

## 3.3 Neural interactions and concluding remarks

Place, grid, and head direction cells are considered the three most important cells in the Hippocampus and Entorhinal Cortex. Their characteristic spatial responses and their predominant role in the formation of spatial knowledge and during navigational purposes brought them the name *big three* [131]. However, not only their interactions are complex and insufficiently understood. It is also unclear which afferents these cells, and grid cells in particular, receive. Are they driven mostly by external input or is the neural activity sustained due to recurrent activity? On the one hand, grid cells require excitatory drive from the Hippocampus [28]. On the other hand though, lesion studies, removal of Theta, or the removal of visual stimuli indicate that grid cells are significantly driven by external input [55, 189, 317].

Many models for place and grid cells use the latter to generate the first [42, 330]. Grid cells are one synapse upstream of place cells and are therefore considered a likely candidate for driving input to the place cell system. Given their multiple spatial scales, they provide sufficient information for a unique spatial localization in comparably large environments. Theoretical investigations are in support of this view [233, 335, 375]. However, the following recent discoveries challenge this assumption. Not only were stable place fields found in pre-weanling rats, whereas grid cell activity could not be found in this early stage of development [204, 382]. In fact, grid cells appear abruptly, but certainly after place cells [236, 341, 381, 383]. Furthermore, both head direction as well as boundary cell activities were reported to demonstrate adult-like behavior early during development and temporally before the appearance of grid cells [27]. Both of which are sufficient to model place cell activity in the absence of grid cell firing [13]. Hence, it seems more likely that grid cells provide supportive data for place cell firing instead of being their driving input.

The interaction between grid and boundary cells are even less understood. It will

be addressed partially in Part II, where boundary information is assumed to form a suitable input space for spatial disambiguation. As was already demonstrated in a boundary vector cell model of place cell firing [13], boundary vectors carry sufficient information to uniquely identify arbitrary locations in complex environments except for symmetries. It is proposed in this thesis that one of the driving afferents to grid cells are boundary cells. Hence, it is suggested to add boundary cells to the *big three*, essentially forming the *big four*.

**Part II**

**Multi-Transition Theory**  
**with an application to**  
**Neural Spatial Navigation**



## Chapter 4

# The motivation for a novel theory

Edward Tolman challenged the scientific community in 1948 by proposing his cognitive map theory [354]. At that time, the behaviorist opinion was that the mammalian brain learns stimulus response patterns for navigational purposes. He opposed this view and suggested that the brain acquires representations of the surrounding environment and performs complex information processing tasks. The already existing interest in memory formation and spatial navigation increased further after his proposal, one of the main reasons being that spatial navigation was – and still is – considered to be a stepping stone to understand higher cognitive functions [255, 256].

The subsequent decades disclosed several types of neurons which were clearly in favor of Tolman’s theory [253, 257]. The seminal work by John O’Keefe and John Dostrovsky [265], for instance, uncovered *place cells* in the Hippocampus of rats whose spiking activity was correlated with only one or a few locations in an environment. Afterwards, other types of neurons were reported which had strong correlations to spatial information. For example, head direction cells fire only when an animal looks towards a specific direction relative to the environment, thereby exposing a functionality similar to a compass [290, 343]. One of the more recent discoveries are *grid cells* in the rodent mEC [135], thought to be fundamental for spatial navigation [118]. In contrast to place cells, they express activity in several, regularly arranged locations of an environment. The reason for and purpose of their distinguished hexagonally arranged fields of activity remain elusive, though. Today, most researchers are in favor of one of two explanations. The cells are believed to either perform path integration, or to contribute to localization. However, both interpretations have significant issues.

Therefore, this chapter motivates the necessity of a novel model and fundamental theory for spatial navigation in the rodent brain. For this purpose, a condensed overview of models for grid cells is presented, and issues are stated which are considered unsolved by existing models. Subsequently, the concrete research question leading to the development of a novel theory is stated and influential related work is noted.

### 4.1 A brief tour of models for grid cells

The discovery of grid cells led to a vast array of computational models to explain their peculiar properties, reviewed by Giacomo et al. [122], Zilli [398], and most recently by Shipston et al. [324]. Oscillatory interference models, for instance, use

interactions of oscillators to generate the characteristic, spatially correlated response of grid cells [40]. These oscillators in turn depend on the Theta rhythm, an oscillatory frequency that is measurable in LFP signals [48]. Other models were suggested in which grid cell activity forms by principles of self-organization due to converging input from place cells [125, 196, 338]. Finally, CAN models rely on recurrent collaterals to generate a Turing pattern [70, 116, 359]. The emerging hexagonal pattern within the network exhibits grid like firing responses and can be used for path integration [39]. Support for the latter models was reported recently by Yoon et al. [391], who found indirect evidence for local attractor dynamics in neural recordings of mEC. Other evidence however, showing that grid cell firing in the rodent brain depends on the Theta rhythm, is in favor of the oscillatory inference models [189]. Despite their differences, most models generate neural responses which are phenomenologically close to biological data, and are based on the assumption that grid cells either perform path integration or localization. However, it is considered doubtful if a hexagonal arrangement of fields of activity is a suitable approach for path integration. The regular distribution of fields introduces ambiguities which have to be resolved using multiple scales of grid cell responses [99].

Theoretical investigations examined the functional characteristics of grid modules, their responses, and the impact of scale discretization during localization. It was proved mathematically not only that grid cells can perform error correction [334]. Furthermore, the discrete scales in combination with the tessellating property of grid cells form an optimal encoding of two dimensional space [375]. In the latter analysis a scale increment of  $\sqrt{2}$  arises naturally from optimality constraints and is hence in concordance with measurements of real grid cells [336]. The emerging distributed code grossly outperforms place cells and can be used in Bayesian inference models to estimate the current location [233, 335]. The results generalize to higher dimensional spaces, which is of interest to researches studying grid cells in other animals, e.g. Egyptian fruit bats [388]. In addition, it was recently suggested that neural representations should exhibit discretized scales if they were to achieve efficient encodings for planning with minimal description lengths [240]. However, the theoretical investigations fall short on explaining the results in biologically plausible systems.

The theoretical models point to an important issue, called *problem of double redundancy*. Why are there two distinct systems for location encoding, namely place and grid cells, considering that maintenance of neural networks is energetically expensive and leads to evolutionary pressure [262]? The grid response is theoretically superior to place codes for spatial inference [233]. Furthermore, distributed codes improve information storage capacities in associative memories [180, 269, 271]. Hence a distributed state representation, such as in a multi-scale grid cell code, outperforms place cells in representational capabilities and memory capacity if both place and grid cells were to encode for spatial location.

There is another issue besides the problem of double redundancy. Only few models for spatial navigation involving both place and grid cells explicitly consider the temporal aspect of sequences of places [133]. Most literature either ignores temporal characteristics that are not consequences of neuronal dynamics completely, or pays just little attention to them. However, the HF is long known to be a critical area for episodic memory [167, 319, 358]. Furthermore, time plays a crucial role in memory

formation in the Hippocampus as well as in the operation of grid cells [96, 193, 309]. Conclusively, there is a missing link between the temporal nature of a sequence of places and its spatial correlate which is not yet addressed in current models of grid cells.

## 4.2 Core question and related work

The issues with the existing models for grid cells lead to the following question. Are grid cells the result of a distinct operation which is related to and in support of place cells? More specifically, does the characteristic spike response of these neurons, which are considered fundamental for spatial navigation, and their spatial scales emerge in a mathematical framework for trajectory planning which links space and time? The question as well as the resulting framework to answer it are rooted in the observation that one of the primary functions realized in the Hippocampus and EC is goal-directed navigation [12, 118, 251]. This in turn requires learning and retrieval of temporal as well as spatial transitions between places.

A related idea concerning spatial transitions in the Hippocampus was previously suggested by Cuperlier et al. [73, 74, 146]. However, the authors ignored multiple scales or optimality of the encoding. Furthermore, sequences and transitions were not rigorously defined in a proper mathematical framework. Other studies explored sequence learning in episodic-memories using STDP but did not touch on the subject of grid cells [142, 143].

A neuron model which shares similarities on the level of the network dynamics and organization is the rate adaptation model presented by Kropff et al. [196]. In that model, fatigue dynamics modulate the firing rate of simulated neurons. Using spatially modulated neurons as input, i.e. place cells, the network self-organizes grid cells with hexagonal firing fields in Euclidean space. The model was later extended to include local recurrent competitive dynamics in a network of grid cells [325]. Thereby, grid responses were successfully stabilized such that grid fields shared their alignment and hence formed coherent grid modules. The computational purpose of grid cells remains elusive in this model, though. Furthermore, the time required for the self-organization process is biologically implausible.

The fundamental concept of the model presented by Kerdels et al. is close to the results and methods presented in this thesis [178]. Specifically, the model by Kerdels et al. assumes that dendritic computations lead to hexagonal firing fields of grid cells. Furthermore, the dendritic computation performs clustering of the input in form of Voronoi cells similar to the results presented in Chapter 7. Likewise the extension to the rate adaptation model by Si et al. [325], the model by Kerdels et al. presents a computational method based on local competitive dynamics to align multiple grid cells. Similar ideas were used in the following chapters. However, the model of Kerdels et al. requires a periodic input space and performs localization, whereas the model of grid cells presented in this thesis encodes transitions between locations.

The proposed novel theory for goal-directed navigation in the entorhinal-hippocampal loop is derived using tools from symbolic and propositional logic, computability theory, and graph theory. Hence, the resulting resemblance to automata or Reinforcement Learning is certainly not coincidental. So far, only few authors examined the logic

of computations in neural networks with propositional logic or transition systems, e.g. [136, 297, 344]. However, these techniques are provably powerful to understand and analyse distributed systems [218, 361]. Here, they are used to derive the formal logic and, later, define the algorithmic concepts.

The formal theory of transition coding presented in this thesis and the development of the models spans all three levels of analysis proposed by David Marr [230, p.25]. He suggested to examine neural systems on the levels of computational theory, representation and algorithm, and hardware implementation. Furthermore, it borrows the coherency and consistency constraints proposed in the same work.

# Chapter 5

## On Multi-Transition Systems

Any mobile animal requires the capability to travel to goal locations if it wants to survive. When resting at a single location for long periods it may fall subject to predators. Given a starting point, the task is therefore to compute one or multiple possible trajectories which traverse intermediate locations until a target is reached.

Each trajectory can be augmented by sub-goals. For instance when returning home from the feeding site the animal wishes to visit a waterhole, or it may want to avoid an area in which it saw its predator. Hence, a trajectory can consist of sub-trajectories which were learned previously and associated with positive or negative rewards, for instance with the help of some reinforcement learning process. Generally speaking, sub-goals are a recursive evaluation of the path planning operation. It is therefore sufficient to focus on planning trajectories to a single goal for the time being.

The following formal system, named MTT, is presented in the context of spatial navigation. However, it is kept deliberately abstract. It is postulated that it also applies to other domains in which the storage and retrieval of sequence points are important. The concatenation of phonemes to produce words or the computation of trajectories while grasping an object are just two of many examples in which sequences and transitions are required. The formalism and notation are loosely inspired by CSP proposed by Tony Hoare [147], and space-time descriptions of distributed processes as used by Leslie Lamport [203].

### 5.1 Alphabets and the computational logic of path planning

A trajectory is a consecutive sequence of points in some space. More abstractly, it is a succession of symbols that are associated with a meaning, e.g. locations, numbers, or events. For instance if a person goes from the living room  $A$  through the door  $B$  to the car  $C$ , the sequence of locations would be denoted  $A, B, C$ . Likewise, the symbols could be assigned to the *event of perceiving* the room, door, or car, respectively. Therefore, when working with abstract symbols, the underlying meaning of a symbol is allowed to change, while the computation remains the same. The concept of symbols and sequences is formalized more rigorously within the following axiomatic system.

**Definition 1** (Alphabet and sequence). *An alphabet  $\Sigma$  is a finite set of symbols. A sequence (or word) is an ordered tuple of symbols  $\sigma_i \in \Sigma$ , formally denoted as*

$(\sigma_0, \sigma_1, \dots) = w \in \Sigma^+$ , where  $+$  is the Kleene plus operator.

The function  $\eta : \Sigma \rightarrow \mathbb{N}$  called index of assigns each symbol in  $\Sigma$  a unique natural number  $n \in \mathbb{N}$ .  $\eta^{-1}$  is its inverse.

The definition is not restrictive on the succession of symbols. For instance, the sequence  $AAA$  could be generated. It should be considered degenerate from the perspective of trajectory planning, though. For instance, prey that goes from one hideout to another should not rest at an open space just because its path planning system tells it to. It could fall victim to a nearby predator. This leads to the following constraints.

**Axiom 1** (Non-stationarity). *A sequence is called non-stationary if any two successive symbols  $\sigma_i$  and  $\sigma_{i+1}$  are distinguishable, i.e.  $\sigma_i \neq \sigma_{i+1}$ .*

One may argue that non-stationarity limits general capabilities that lie beyond path planning, e.g. if one symbol is required multiple times. However the definitions presented here could easily be extended. For instance by having multiple different symbols with the same meaning, but containing a potentially hidden contextual information for disambiguation. Such an extension expresses principal similarities to Hidden Markov Models.

To capture the *temporal* and *directional ordering* of sequences typographically, the arrows  $\rightarrow$ ,  $\nrightarrow$ ,  $\rightsquigarrow$ , and  $\not\rightsquigarrow$  are used in combination with symbols. In the example from above, the arrow  $\rightarrow$  reads as "immediately happens before" such as in  $A \rightarrow B$  ("A immediately happens before B"). " $\nrightarrow$ " indicates that there is no direct succession of two symbols, e.g.  $A \nrightarrow C$ . Still, there exists a *path* from A to C denoted as  $A \rightsquigarrow C = A \rightarrow B \rightarrow C$ . Conversely,  $C \not\rightsquigarrow A$  in the given example.

**Axiom 2** (Temporal coherency). *Let  $w$  be a sequence of  $N$  symbols  $\sigma_i, i \in \{0, \dots, N\}$ .  $w$  is said to be temporally coherent (or simply coherent) if and only if  $\sigma_i \rightarrow \sigma_{i+1}, \forall i \leq N - 1$ .*

In other words, a temporally coherent sequence must consist of distinguishable successors that follow each other, gaps are not allowed. They could lead to indeterminate behavior of an animal, e.g. getting stuck because it does not know how to continue.

**Axiom 3** (Validity). *A sequence  $w$  is valid or acceptable if it is both non-stationary and temporally coherent.*

The *computational logic* on the highest level of path planning can now be expressed symbolically. Let an animal reside at an initial location  $\sigma_s$  with the goal to navigate to a target location  $\sigma_t$ . Assume that there exists such a path and that this knowledge is available somehow in the animal's memory. Therefore path planning corresponds to expanding a path  $\sigma_s \rightsquigarrow \sigma_t$  into a valid sequence of symbols. In other words,  $\sigma_s \rightsquigarrow \sigma_t$  is the input to a program which yields as output a coherent non-stationary sequence  $\sigma_s = \sigma_0 \rightarrow \sigma_1 \rightarrow \dots \rightarrow \sigma_N = \sigma_t$ .

## 5.2 Universal Multi-Transition Systems

How can the animal compute a valid expansion of  $\sigma_s \rightsquigarrow \sigma_t$ ? Are there multiple ways to reach  $\sigma_t$  and, if so, which one is to prefer? To be able to answer these

questions it is necessary to examine the transitions between successive symbol and, most importantly, their representation.

Inherent in the arrow notation  $\rightarrow$  is a mathematical object which encodes relations between other mathematical objects. For instance  $A \rightarrow B$  denotes a tuple  $(A, B)$ . In Reinforcement Learning (RL) this is typically expressed as a transition function  $\tau : \Sigma \times R \rightarrow \Sigma$ , i.e. a function which maps a symbol (or state) to another symbol given some general rules of movement  $R$ .

Here, a representation for transition systems used in theoretical computer science is borrowed and extended to allow multiple transitions. Consider the example from above (room  $A \rightarrow$  door  $B \rightarrow$  car  $C$ ), but this time the room has an additional door  $D$  which also leads to the car. Hence there exist two possible trajectories from the room to the car, i.e.  $A \rightarrow B \rightarrow C$  and  $A \rightarrow D \rightarrow C$ . This is captured in the following definitions and notations.

**Definition 2** (Transition system, set, bundle, and point). *A MTS  $\mathcal{M}$  is the pair*

$$\mathcal{M} = (\mathcal{P}(\Sigma), \Pi) \quad (5.1)$$

where  $\mathcal{P}(\Sigma)$  is the power set of  $\Sigma$  describing all possible configurations. The set  $\Pi$  is called transition set and contains other sets, called transition bundles  $\pi_i$ . In turn, transition bundle  $\pi_i$  contains tuples of the form  $\tau_i^k : \Sigma \times \Sigma$  called  $k$ -th transition point of  $\pi_i$ , or simply transition.

A configuration  $\Omega \subseteq \mathcal{P}(\Sigma)$  corresponds to the set of active symbols, and any combination of symbols can form a configuration. In other words, a configuration is the set of propositional symbols which are *true*. For instance, if  $A$  is in the configuration  $\Omega$ , then  $A$  is *true*. Furthermore, the following notations will be used for any transition set  $\Pi$ , bundle  $\pi$ , and point  $\tau$ .

1. A transition  $\tau$  from  $A \in \Sigma$  to  $B \in \Sigma$  can be written either  $(A, B)$  or  $(A \rightarrow B)$ .
2. Transition  $\tau = (A \rightarrow B)$  is said to be *defined* for  $A$  and *leads to*  $B$ , denoted  $A \prec \tau$  and  $\tau \succ B$ , respectively. The notation is transitive to bundles and sets, i.e.  $A \prec \pi \Leftrightarrow \exists \tau \in \pi, A \prec \tau$  and  $\pi \succ B \Leftrightarrow \exists \tau \in \pi, \tau \succ B$ .
3. A transition bundle  $\pi$  can be written as the tuple  $\pi = (S, T)$  with start symbols  $S = \{\sigma | \sigma \prec \tau, \tau \in \pi\}$  and target symbols  $T = \{\sigma | \tau \succ \sigma, \tau \in \pi\}$ .

The transition system allows logical deduction when symbols are used as propositions. If a symbol  $A$  and a transition  $(A \rightarrow B)$  are both true,  $B$  is implied to be *true*. This can be written compactly as  $A \wedge (A \rightarrow B) \Rightarrow B$ , where  $\wedge$  reads as *logical and*. Thus,  $A$  forms a precondition to  $(A \rightarrow B)$  and  $B$  its conclusion, given the precondition is true.

Despite the term *precondition*, it is crucial to note that the concept of temporal ordering of events is *not* applied to the formalism when analyzed logically. Hence, temporal ordering of evaluations can be made more explicit with the following definitions.

**Definition 3** (Transition evaluation). *A configuration  $\Omega \subseteq \mathcal{P}(\Sigma)$  of a MTS  $\mathcal{M}$  is evaluated according to the functions*

$$F_{\mathcal{M}} : \Omega, \Pi \mapsto \cup_i f_{\mathcal{M}}(\Omega, \pi_i \in \Pi) \quad (5.2)$$

$$f_{\mathcal{M}} : \Omega, \pi \mapsto \{\sigma_l | \sigma_k \in \Omega, \sigma_k \prec \pi, \sigma_l \text{ is true in } \pi\}, \quad (5.3)$$

with the shorthand notations  $\Pi(\Omega) := F_{\mathcal{M}}(\Omega, \Pi)$  and  $\pi(\Omega) := f_{\mathcal{M}}(\Omega, \pi)$ , respectively.

On evaluation, a transition set  $\Pi$  therefore yields a set of all symbols which reduce to *true* given the current configuration  $\Omega$  and the transition bundles which are defined for symbols in  $\Omega$ . Likewise, a bundle  $\pi_i$  returns a set of all symbols which are *true* given  $\Omega$  and its transition points. For the example above, the configurations correspond to any combination of the rooms  $A, B, C, D$ , and

$$\Pi = \{\pi_0 = \{\tau_0^0, \tau_0^1\}, \pi_1 = \{\tau_1^0, \tau_1^1\}\}, \quad (5.4)$$

where the transition points are defined according to

$$\tau_0^0 = (A, B), \tau_0^1 = (A, D), \tau_1^0 = (B, C), \tau_1^1 = (D, C). \quad (5.5)$$

Indexes will be omitted or reduced to a singular subscript index if they are not relevant or can be inferred from context, e.g.  $\tau_k = \tau_i^k$  if  $i$  is obvious.

Note the relationship to the vector representation of RL or other automata based notations. There, state-action pairs are commonly denoted by a transition matrix  $\mathbf{R}$ , for which  $\Pi$  is the analog. If multiple co-active symbols and transition results are disallowed, then any  $\pi_i$  corresponds to a vector of  $\mathbf{R}$  and  $\tau_k$  to its  $k$ -th entry.

Furthermore the definition can be broadened to allow non-deterministic, probabilistic transitions. The result resembles a Markov Decision Process extended to multiple active states. It should therefore be possible to study existing probabilistic methods such as the Forward or Viterbi algorithms with the notation presented here.

The somewhat abusive notations of a set and function have several benefits. Parallelism and branching are compactly enclosed within the notation, similar to matrix notation. The example above can be expressed as the recursive program  $\Pi(A \rightsquigarrow C) = \Pi(\Pi(A))$ , for which  $C \in \Pi(A \rightsquigarrow C)$  holds, regardless if one symbol can lead to multiple other symbols. Consider assigning each configuration and evaluation of  $\Pi$  a time-stamp. At time  $t_0$  the configuration of  $\mathcal{M}$  is  $A$  and thus the transition evaluation is  $\Pi(A)$ . At time  $t_1$  the pair is  $\{B, D\}$  and  $\Pi(\{B, D\})$ . In other words,  $\Pi$  defines the transitions in a parallel state machine in which multiple states can be active at the same time, and transitions from one state to another are handled by  $\pi$ .

The *bundling trick*, i.e. introduction of transition bundles  $\pi$ , is essential to study optimal transition encoding in neural networks. For instance, the trick allows to reason about the physical implementation, computational logic, and storage requirements of transitions in real neurons such as place and grid cells. It also provides a way to investigate the response fields of neurons representing transitions. Both is not directly possible otherwise.

Suppose that it is expensive to store only a single transition point in a transition bundle due to some reason. Furthermore, assume that it is cheap to add more transition points to one transition bundle. For instance, the energetical production cost of a whole neuron is assumed to exceed the construction of an additional dendritic or axonal branch. Hence, the number of transition neurons required is related to reducing the overall cost. In other words, the optimal number of transition neurons with respect to this cost is achieved by maximizing the number of transition points stored within a minimal number of transition bundles. However, transition bundles are subject to the following theorem.

**Theorem 1.** *Let  $\sigma \in \Sigma$ ,  $\mathcal{M}$  a MTS on the alphabet  $\Sigma$ ,  $\Pi$  the corresponding transition set, and  $\pi = (S, T)$  a transition bundle.  $\mathcal{M}$  can generate coherent non-stationary sequences if and only if the following conditions hold.*

1.  $\sigma_k \prec \pi \implies \pi_k \not\prec \sigma_i$ .
2.  $\pi \succ \sigma_l \implies \sigma_l \not\prec \pi$ .

*Proof.* 1. From Axiom 1 it follows immediately that any transition  $\pi$  which is defined for  $\sigma_k$  and leads to  $\sigma_k$  violates the non-stationarity condition. 2. Without loss of generality, consider the three symbols  $\sigma_0, \sigma_1, \sigma_2 \in \Sigma$  such that  $\sigma_0 \rightarrow \sigma_1 \rightarrow \sigma_2$  but  $\sigma_0 \not\rightarrow \sigma_2$ . This is expressed in the transition points  $\tau_0 = (\sigma_0, \sigma_1)$  and  $\tau_1 = (\sigma_1, \sigma_2)$ . Assume  $\tau_0$  and  $\tau_1$  are bundled in  $\pi$ . Given  $\sigma_0$  and  $\pi$  are *true* (or active). It follows from  $\sigma_0 \wedge \tau_0 \Rightarrow \sigma_1$ . However,  $\sigma_1 \wedge \tau_1 \Rightarrow \sigma_2$  and thus  $\sigma_0 \wedge \pi \Rightarrow \sigma_2$ . Therefore,  $\pi$  tells that  $\sigma_0 \rightarrow \sigma_2$  is feasible which contradicts the assumption and the coherency constraint.  $\square$

In other words, the input and output sets  $S, T$ , respectively, of a transition bundle  $\pi$  are mutually exclusive, i.e.  $S \cap T = \emptyset$ .

**Corollary 1.** *The input set  $S_i$  of a transition  $\pi_i$  is singleton for a minimal universal  $\mathcal{M}$ .*

*Proof.*  $\mathcal{M}$  is said to be *minimal* if there exists only one  $\pi_i$  for any  $\sigma_k$ , i.e.  $\sigma_k \prec \pi_i \Rightarrow \sigma_k \not\prec \pi_j$  for any  $j \neq i$ . Any transition between two symbols  $\sigma_k, \sigma_l$  are possible in a *universal*  $\mathcal{M}$ . Therefore  $\sigma_k \prec \pi_i$  and  $\pi_i \succ \sigma_l, \forall l \neq k$ . According to Theorem 1,  $\sigma_l \not\prec \pi_i, \forall l \neq k$ .  $\square$

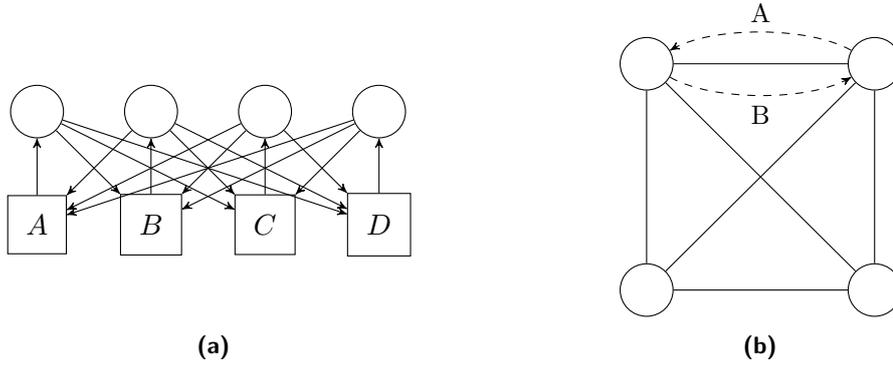
**Corollary 2.** *Let  $\Sigma$  be an alphabet of size  $M$ ,  $\Pi$  a transition set of size  $N$  for a minimal universal  $\mathcal{M}$ . Without further restrictions,  $M = N$ .*

The following constructive proof is rooted in graph theory. As is common in computer science, abstract rewriting systems and transition systems can be represented as directed graphs.

*Proof.* The transitions from one symbol to any other can be represented by a bipartite directed graph  $\mathcal{G}$ . Each node representing one symbol is connected to one node corresponding to a transition to other symbols. Each transition node is connected with a directed edge to any other symbol node that can be reached by the initial symbol.

The bipartite directed graph can be reduced by replacing any path across a symbol by a single directed edge. Subsequently, any pair of directed edges connecting a pair of transition is replaced by an undirected edge. The result is an undirected graph.

Minimizing the number of transition bundles requires to bundle as many transition points in one bundle as possible. However, Theorem 1 states that  $S_i \cap T_i = \emptyset$  for any  $\pi_i$ . This means that only transition points which are not connected by an edge in the graph can be bundled. The number of transition bundles required to fulfill Axiom 1 and Axiom 2 therefore corresponds to the chromatic number of the graph, i.e. the number of different colors required in the graph coloring problem [69]. For a minimal universal  $\mathcal{M}$ , the resulting graph is complete for which the chromatic number equals the number of vertices.  $\square$



**Figure 5.1 – Example of symbols and transitions as a bipartite graph and its reduction to a undirected graph. (a)** Squares represent symbols and circles represent transitions. **(b)** Each pair of directed edges from the original graph of which lead from one transition to another over only a single symbol (dashed arrows) are replaced by a singular undirected edge.

An example of a bipartite transition graph corresponding to an alphabet of four symbols is shown in Figure 5.1a. The figure depicts four symbols, each as a square node, and each corresponding transition as a circle. Arrows depict the direction of the transition. The bipartite directed graph is reduced to an undirected graph in Figure 5.1b.

In conclusion of the proof for Corollary 2, the minimal number of transition bundles can be studied by analysing the corresponding graph coloring problem. Note though that the procedure may not necessarily hold if a sequence contains directional restrictions, i.e. if  $A \rightarrow B$  is valid but  $B \rightarrow A$  is not. The study of this issue is left for future work though, and from now on all inverse transitions are considered feasible.

### 5.2.1 Interim observations and implications for neural networks

The results imply certain constraints for any implementation of a universal MTS  $\mathcal{M}$ . The system requires at least as many transition bundles  $\pi_i \in \Pi$  as there are symbols  $\sigma_i \in \Sigma$  to unambiguously store and retrieve any sequence that it encounters and is subject to the coherency and non-stationarity constraint. Such a memory system has to cope with the possibility of a direct transition from any symbol  $\sigma_i$  to another symbol  $\sigma_j$ . In addition, the input space for which the system is used has to be sampled densely to learn all of these transitions. There is no functionality within the system to learn or re-generate transitions that have not been observed previously.

Four implications for a neural implementation of  $\mathcal{M}$  follow promptly:

1. A neuron, or generally speaking a neural state, which represents a transition bundle will co-activate with its associated input symbol. In other words, it will *inherit* the activity field of the input symbol.
2. If implemented as a recurrent neural network, the recurrent connectivity of a transition bundle neuron to symbol neurons will show correlation only with target symbols and must decorrelate from its input.
3. The neural system needs to implement a *logical and* to fulfill the precondition constraint, for instance in form of hetero-synaptic connections.

4. Ignoring additional context, there are as many symbol neurons as there are transition bundle neurons.

Also worth observing, the symbols and transition bundles of a minimal universal  $\mathcal{M}$  form a discrete topological space with a distance metric  $d$ . According to Theorem 1, any symbol  $\sigma_j$  is in the *deleted neighborhood*  $N_- = \Sigma \setminus \sigma_i$  of any other symbol  $\sigma_i$ , i.e.  $\sigma_j \in N_-(\sigma_i)$ . Consequently  $d(\sigma_i, \sigma_j) = 0 \Leftrightarrow \sigma_i = \sigma_j$  and  $d(\sigma_i, \sigma_j) = 1$  otherwise.

## 5.3 Sequences in continuous metric space: Emergence of grid cells

Space is not a discrete topological space on the perceptible level of an animal. The distance to any other location is neither a constant, nor is it possible to arbitrarily jump between places. Hence, there is no necessity for a spatial transition system  $\mathcal{L}$  to encode transitions between arbitrary symbols. Rather, only transitions between neighboring points of interest or spatial symbols of a spatial alphabet, formally denoted as  $\delta_i \in \Delta$  to distinguish them from  $\sigma_j \in \Sigma$ , are required.

### 5.3.1 On dense sampling and sphere packing

Although space is continuous, a biological neural network is finite in the number of neurons. Hence the input space has to be sampled.

Assume that any point in an  $n$ -dimensional metric space  $\mathcal{D} = (M, d)$  has a unique signature of sensory readings. This sensory space itself is continuous and therefore differentiable. Mathematically, this corresponds to the coordinates of a point and the distance between two points  $\mathbf{p}_1, \mathbf{p}_2 \in M$  can be expressed by the metric  $d(\mathbf{p}_1, \mathbf{p}_2)$ . Points and vectors will be typeset in boldface to distinguish them from symbols.

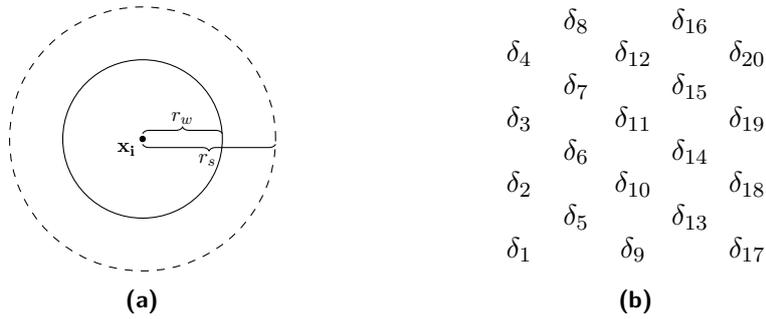
Sampling from the sensory space can be considered one of two related issues. On the one hand it corresponds to the sphere packing problem [68,202]. The radius  $r_s$  of a sphere corresponds to the resolution of the sampling process. The sphere packing and sphere covering problems were studied extensively and have important applications in optimal coding theory and especially in error-correcting codes [68,103,209]. On the other hand, the issue can be expressed in terms of sampling from a band-limited signal in higher dimensional spaces according to the Petersen-Middleton theorem [281]. Both methods, sphere packing and sampling theory, yield optimal results in lower dimensional spaces when a regular lattice is used [263]. Higher dimensions are difficult to prove and results were reported only recently [65,66,367]. However, a hexagonal lattice of sampling points or circles is ideal for the two dimensional case [281]. In the case of a three dimensional signal, sphere packing is optimal [263,281].

Conclusively, it is assumed that there exists an ideal sampling process which yields a hexagonal pattern of spatial symbols  $\delta_i$  of the spatial alphabet  $\Delta$ . This assumption will be used during the analysis of optimal transition representation in the following section.

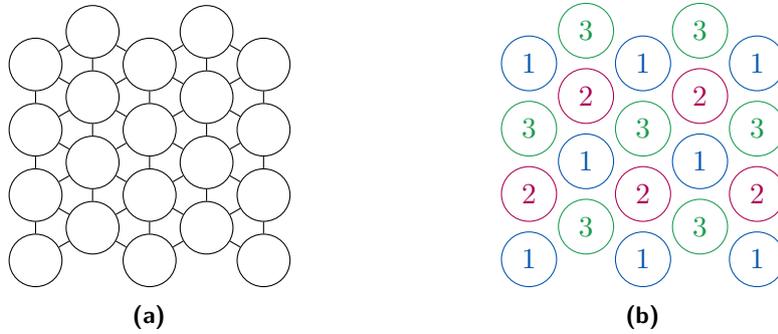
### 5.3.2 Spatial neighborhood transitions and grid cells

**Definition 4** (Spatial symbol, support, and assignment). *Let  $\delta_i \in \Delta$  be spatial symbols according to an ideal sampling process in a metric space  $\mathcal{D} = (M, d)$ . Each*

## 5. On Multi-Transition Systems



**Figure 5.2 – Spatial symbol and densest packing of symbols in two dimensions.** (a) Two dimensional depiction of a spatial symbol  $\delta_i$  centered at  $\mathbf{x}_i$ . The *support* of  $\delta_i$  corresponds to the ball with radius  $r_s$ . Any point within the ball of radius  $r_w$  is said to be *assigned* to the symbol. (b) Example of the densest spatial symbol arrangement in two dimensions.



**Figure 5.3 – Transition graph in a metric space and its graph coloring.** (a) Transition graph between neighboring symbols for Figure 5.2b. (b) One solution to the graph coloring problem on the transition graph of (a), numbers correspond to colors.

$\delta_i$  is thus centered at a  $\mathbf{x}_i \in M$ . A point  $\mathbf{p} \in M$  enables  $\delta_i$  if it is within the support of  $\delta_i$  given by the open ball  $B_{i,s}$  of radius  $r_s$ , i.e.  $B_{i,s} = \{\mathbf{p} \in M | d(\mathbf{x}_i, \mathbf{p}) < r_s\}$ . The point  $\mathbf{p}$  is assigned to the closest  $\delta_i$ , i.e.  $\delta_i$  for which  $d(\mathbf{x}_i, \mathbf{p})$  is minimal. Given two adjacent symbols  $\delta_i, \delta_j$ , then  $r_w = \|d(\mathbf{x}_i, \mathbf{x}_j)\|/2$ , describing a ball  $B_{i,w}$  of radius  $r_w$ .

The parameters  $r_w$  and  $r_s$  are constant for all symbols and  $r_w \leq r_s$ . The definition is visualized in Figure 5.2a, and the balls around  $\mathbf{x}_i$  can be understood as follows.  $B_{i,s}$  defines the receptive field of symbol  $\delta_i$ , whereas  $B_{i,w}$  denotes the area in which  $\delta_i$  is considered the *winning symbol* if multiple symbols were enabled. In neural networks this translates either to the firing rate or the precise spike timing, depending on the mechanism that is used for neural coding.

**Theorem 2.** Let  $\mathcal{D} = (M, d)$  be a metric space. Let  $\mathcal{L} = (\mathcal{P}(\Delta), \Gamma)$  be a minimal transition system on  $\mathcal{D}$  such that the countably finite alphabet  $\Delta$  corresponds to the densest optimal covering with respect to  $r_w$ .

1. The number of transition bundles  $\gamma_i \in \Gamma$  is constant.
2. The occurrence of any transition bundle  $\gamma_i$  is periodic.

*Proof.* The corresponding graph coloring problem introduced in Section 5.2 is used to prove the theorem in two dimensions. The densest arrangement of spatial symbols is depicted in Figure 5.2b and forms a hexagonal lattice. Transitions between symbols are only possible between adjacent symbols. In other words, all symbols  $\delta_j$  that are at most  $2r_w$  apart from  $\delta_i$  form a *local transition group*. Consequently, the corresponding transition graph extracted according to the method described in the proof for Corollary 2 is not complete, i.e. only the local transition group is connected. The transition graph for two dimensions is depicted in Figure 5.3a. The chromatic number of the resulting graph is 3 and the occurrence of colors is periodic as depicted in Figure 5.3b.  $\square$

It is conjectured that the proof will hold in higher dimensions.

## 5.4 Discussion and remarks on the biological plausibility

MTT defines symbols and transitions both for purely episodic as well as spatial information, the first in case of a universal MTS. Likely neural candidates for symbols are place cells of hippocampal area CA3. As these cells are relevant for both episodic as well as spatial information processing [229, 251, 265, 319], it is proposed that they form encoders of spatio-temporal symbols. Similar to the model by Barry et al. [13], it is proposed that place cells form primarily on spatial afferents. However, it is further proposed that the meaning of a symbol in a universal MTS, and hence of a place cell, can be extended such that it integrates additional non-spatial afferents due to the observation of conjunctive place cells [225, 247]. The preceding analysis of a MTS remains unchanged by this extension.

So far it is unclear if inter-neurons in CA3 could represent temporal transitions, or if temporal transitions are rather encoded in the place cells of CA1. Recurrent collaterals in CA3 are in favor of the first assumption [11, 205]. Recurrent collaterals from CA1 to CA3 have not been described often enough in support of the second possibility. However, it is possible that the recurrent connectivity is not mono-synaptic but is represented by what is known as the *trisynaptic loop*, spanning from CA3 across CA1 to EC, before it arrives back at CA3 [6].

Following MTT and Theorem 2 in particular, grid cells are proposed to represent spatial information in form of transitions between spatial symbols. It is proposed that they not only form on the basis of pre-synaptic spatially modulated input, but also affected by recurrent connectivity from place cells. Both propositions were already observed in real grid cells [28, 55]. From a computer scientific perspective, this setup of interactions can be considered an abstraction layer and will be explored in detail in Chapter 7. To anticipate, place cells are suggested to form a storage mechanism for arbitrary points of a sequence, whereas grid cells encode their spatial transitions and provide spatial neighborhood information of places. In this way, place cells are unaware of spatial relations except via the indirection of grid cells. Consequently, the sensory representation by which grid cells formed in the first place may change over time, but the spatial relationship, and therefore knowledge of potentially neighboring locations, is maintained.

The spatial MTS  $\mathcal{L}$  requires a unique sensory representation to identify singular locations and for the optimal sampling assumption. Furthermore, it is necessary to

detect the change between states. Candidates for a neural representation of such a signature and change between locations are head direction, tactile information, optic flow or generally speaking distal visual cues, and ego-motion. Optic flow was successfully used not only to model grid cell firing characteristics in an oscillatory inference model [291, 292]. It was shown to be sufficient to account for boundary cell responses as well [294]. The latter finding could explain observations in real recordings of grid cells in which they were influenced by the geometry of the environment [198, 199]. Furthermore, optic flow contains sufficient information to extract ego-motion which, in turn, is represented in MEC by speed cells which fire linearly with respect to the animal's speed [195]. Additionally, boundary vector cells were successfully used to encode locations and drive place cell activity in a computational model and thereby discriminate positions [13]. Finally, sensory cues were able to stabilize continuous attractor dynamics in a network model of grid cells [258]. Interestingly, it was reported that grid cells require visual input for their periodic responses [55]. It is thus conjectured that visual input and the boundary vector state provide sufficient information for the encoding of locations and formation of transitions in experimental environments of two and three dimensions.

The optimal sampling process requires a dense representation of the input space. Hence, the tuning curves of neurons pre-synaptic to grid cells are expected to overlap appropriately. It is known from several cortical areas, especially the auditory and visual cortices, that neurons show overlapping tuning curves which are indeed well separated, uniformly cover the input space, and are often organized topographically [34, 95, 157, 339]. In all studies, the amount of overlap depends on the tuning width of the neurons and the number of neurons employed to sample from the space.

On the other hand, the amount of overlap of grid cell firing fields is expected to decrease over time. Consider two adjacent spatial symbols at locations  $x_i, x_j$  in a continuous one dimensional space  $\mathcal{D}$ , encoded in form of two neurons such that their receptive fields cover the distance, i.e.  $r_s = 2r_w$ . Then, the precise relative spike time of the two neurons contains sufficient information to determine the exact location between  $x_i$  and  $x_j$ . When does the transition appear and how to encode it? One may suggest that the transition occurs when the difference between spiking times of the neurons changes sign, i.e. when the location moves from  $r_w$  of one neuron to the next. However, the transition will get activated throughout  $r_s$ , and thereby violating the coherency constraint. Therefore it is suggested that a process exists which will try to maximally separate the tuning curves of the cells depending on experience within an area. Then, the activation of a spatial transition neuron would initially start to correspond with  $r_s$  but shrink to  $r_w$  over time. An effect which was already observed in real recordings from the EC [14, 15]. As a by-product, the response field of a transition neuron will likely reduce to a Voronoi cell. Hence, perfect rotational symmetricity with respect to  $r_w$  will decrease. Another feasible solution to the constraint violation is to associate a transition bundle only with targets outside of  $r_s$ , and ignore any other symbols co-active within  $r_s$ . Other transition cells would then be required to densely cover the set of spatial symbols. Local self-organization principles, for instance such as suggested in the computational model for grid cell formation by Kerdels et al. [178], or attractor dynamics could lead to a coherent representation within one grid module.

Besides preventing violation of the coherency constraint by transition points, the

network has to ensure the unambiguity of the transition bundle given its sensory information. This means that at any location which is uniquely identifiable by sensory information, only the transition bundle which is associated with the corresponding sensory state is allowed to be active. This suggests that the transition network is governed predominantly by local inhibitory recurrences generating a winner-take-all mechanism. In fact, it was already observed that recurrent connectivity in mEC is primarily inhibitory [70]. The local inhibition is required to be fast enough to prevent erroneous activation of a transition unit. Such a temporally quick effect has been observed already in the Hippocampus [86], and it is expected that local recurrences in mEC are equally fast.

A model which yields the general behavior expected from the presented theory was published recently by Widloski et al. The authors proposed a spiking neural network which was driven by spatially modulated input and formed a hexagonal lattice in two dimensions [379]. However, the computational necessity of grid cells was not addressed in the work.

Conclusively, the prerequisites for neural implementations of both temporal and spatial MTS exist. A biologically plausible model for grid cells in continuous metric space is derived in Chapter 6. The algorithmic interactions between temporal and spatial MTS are the subject of Chapter 7.



## Chapter 6

# A neural model of self-organizing grid cells

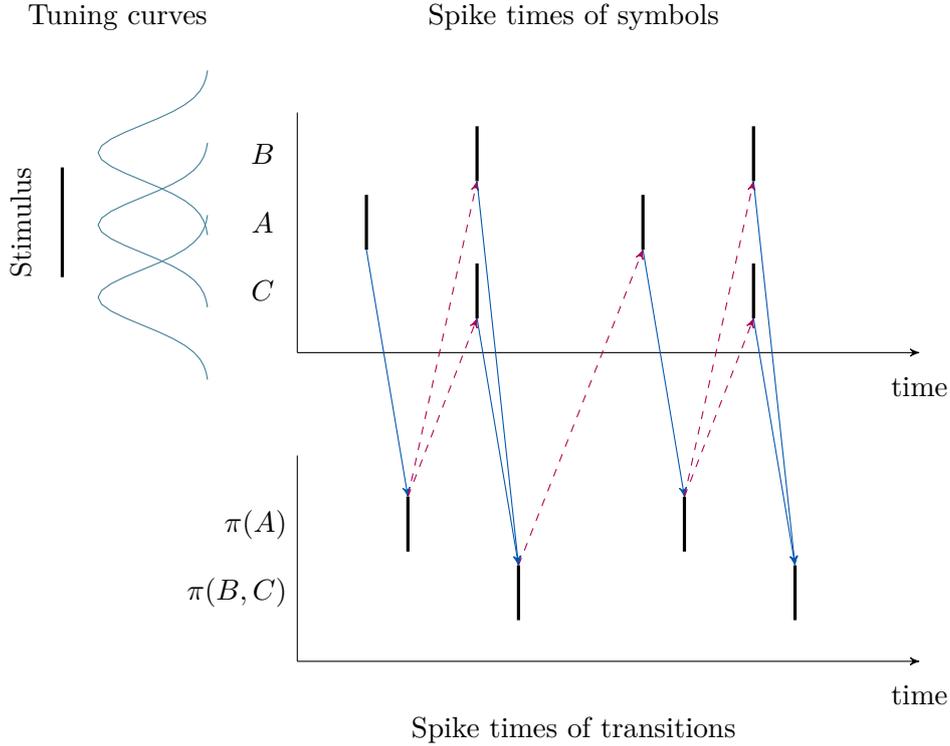
MTT poses significant requirements for real-world implementations of a universal as well as a spatial MTS. For instance, the necessity of transition neurons to decorrelate from their target symbols is critical and will therefore be analyzed with respect to its biological feasibility. Afterwards, a biologically plausible model of a single grid cell is derived mathematically and simulated for various parameters to characterize the system. Finally, the model equations are extended for a network of competitive grid cells. Likewise the single-cell model, the network model will be simulated to characterize its behavior.

### 6.1 Learning to decorrelate input and output

Any transition neuron needs to decorrelate from symbols to which the transition is leading. This requirement, mentioned previously in Subsection 5.2.1, was evaluated in a small network of two recurrently connected neuron populations and one input population and is based on the following hypothesis. The temporal dynamics of individual neurons in combination with an STDP learning rule will correlate neurons of the second population only with neurons of the first population which fired sufficiently early. Conversely though, the recurrent drive from the second population will drive sub-thresholdly active neurons of the first population to their spiking threshold from which the second population will decorrelate. The hypothesis is visualized in Figure 6.1. The left hand side of the top row of the figure shows the stimulus, which is in form of a bar. Three neurons with overlapping and equally spaced tuning curves will collect the input stimulus. The hypothesis is that the neuron on which the stimulus is centered will spike almost immediately (simplified spike-raster plot on top row). Due to recurrent connectivity across transition neurons in the bottom row of the figure, the other two neurons which were activated sub-thresholdly due to only partial overlap of the stimulus and their receptive fields will be driven to their spiking thresholds.

#### 6.1.1 Model overview and implementation details

A basic STDP rule was used to reduce the model complexity [283]. Although more complex learning rules may be used, e.g. the ones surveyed by Markram et al. [226] or Feldman [104], the simple rule is sufficient to demonstrate proof-of-principle.



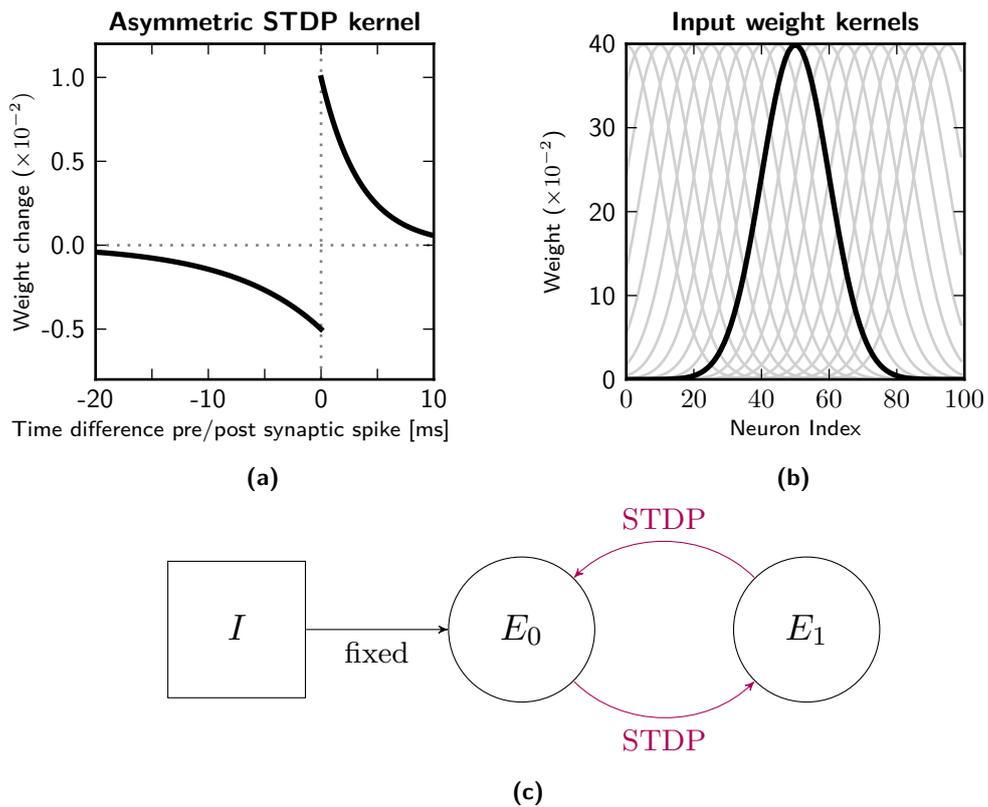
**Figure 6.1 – STDP decorrelation hypothesis.** An input stimulus centered on the symbol  $A$  is presented to a network of neurons representing the symbols  $A, B, C$  with overlapping Gaussian tuning curves. The spike times of the neurons thus depends on the amount of input integration. A recurrently connected network of neurons corresponding to transitions should correlate with a driving input from the symbol neurons (blue solid arrows) and decorrelate from neurons which are in the decorrelation phase of the learning rule (red dashed arrows). Conversely, given axonal latencies that are short enough, the transition neurons should be able to drive symbol neurons to their spiking threshold and thus increase synaptic strength (red dashed arrows) but decorrelate from symbol neurons that already spiked (blue solid arrows).

Formally, the weight evolves according to

$$\frac{dw}{dt}(t) = \eta \begin{cases} A_+ \exp(-t/\tau_+^s) & , t > 0 \\ A_- \exp(-t/\tau_-^s) & , t < 0 \end{cases}, \quad (6.1)$$

where  $t$  is the time difference between pre- and postsynaptic spikes. An asymmetric form with time constants  $\tau_+^s$  and  $\tau_-^s$  was chosen in the range of reported data [23]. Furthermore, the parameters  $A_+$  and  $A_-$  were kept constant. The weights were initialized randomly with mean  $w_0$  and a maximum value  $w_{max}$  was used to limit weight growth. An unusually large learning rate  $\eta$  was used to reduce simulation times. The STDP window according to the parameters is plotted in Figure 6.2a. Asymmetric STDP learning rules have already been demonstrated to be useful in learning sequences in a model of the entorhinal-hippocampal loop [142, 143]. Here, the focus lies on the form of the emerging receptive fields though.

The network consists of one input layer and two layers of excitatory neurons, visualized in Figure 6.2c. The one dimensional input layer consists of stochastic



**Figure 6.2 – STDP kernel and weight distribution and model network layout.** (a) The asymmetric STDP window used for decorrelation learning. (b) Weights from the input to the first layer of the network are modelled as overlapping Gaussians. One curve is plotted bold for visualization purposes only. (c) An input layer of neurons feeds into a first layer of excitatory neurons. The connectivity from the input to the first layer is weighted according to the fixed kernels depicted in (b). The first layer is indirectly recurrently connected by plastic connections with a second layer of excitatory neurons.

Table 6.1 – Model parameters for learning receptive fields.

<b>Simulation parameters</b>		
Numerical integration time window	$dt$	1 ms
Maximum simulation time	$T_{\max}$	10 s
<b>Neuronal dynamics</b>		
Membrane time constant	$\tau_{\text{mem}}^n$	10 ms
Absolute refractory period	$t_{\text{refrac}}^n$	5 ms
Axonal transmission delay	$t_{\text{axonal}}$	3 ms
Spiking threshold	$\theta_{\text{thresh}}$	0.400
Minimum pre-activation threshold	$\theta_r$	0.010
After spike membrane potential	$u_{\text{refrac}}$	-0.100
<b>Synaptic plasticity</b>		
Pre-synaptic spike time constant	$\tau_+^s$	35 ms
Post-synaptic spike time constant	$\tau_-^s$	80 ms
Mean initial weight	$w_0$	0.050
Maximum weight	$w_{\max}$	0.400
Pre-synaptic trace update	$A_+$	0.010
Post-synaptic trace update	$A_-$	-0.005
Learning-rate	$\eta$	1.000

neurons, spiking according to a Poisson process. During simulation, the input layer was pulsed at a regular frequency of 100 ms such that the central neurons are likely to spike. The input neurons are connected to the first layer of the model with constant weights that follow a Gaussian distribution, depicted in Figure 6.2b. The first layer has plastic connections to a second layer of neurons. The weights of the connections are updated according to the STDP learning rule presented in Equation (6.1). The second layer in turn is recurrently connected to the first, also with plastic connections governed by the same rule.

The membrane potential of neurons in both layers is modelled as leaky-integrate and fire dynamics with absolute and relative refractory periods. The membrane state  $u$  of neuron  $i$  in the first layer evolves according to

$$\tau_{\text{mem}}^n \frac{du_i}{dt} = -u_i(t) + \overbrace{H(u_i - \theta_r) \sum_k w_{ik} \delta_{\text{ax}}(t - t^{(k)})}^{\text{recurrent input}} + \underbrace{\sum_j w_{ij} \delta_{\text{ax}}(t - t^{(f)})}_{\text{external input}} + \xi(t), \quad (6.2)$$

until it reaches a certain threshold  $\theta_{\text{spike}}$  at which it will emit a spike. Here,  $H(\cdot)$  is the Heaviside function,  $w_{ik}$  and  $w_{ij}$  are the weights from recurrent neuron  $k$  of the second layer and from input neuron  $j$  to neuron  $i$  of the first layer, respectively.  $\delta_{\text{ax}}(\cdot)$  denotes the Dirac delta and indicates firing times of recurrent ( $k$ ) or input ( $f$ ) neurons, respecting axonal transmission delays  $\text{ax} := t_{\text{axonal}}$ . The term  $\xi(t)$  is additive stochastic noise following a normal distribution which can lead to spontaneous firing of a neuron. The recurrent dynamics embedded in Equation (6.2) allow to push

neurons out of a sub-threshold domain to spiking behavior. The equation for neurons of layer II follow Equation (6.2) except that there is no external input and the Heaviside function is omitted.

After a spike occurred, a neuron will undergo a short absolute refractory period of duration  $t_{\text{refrac}}$  in which it is not able to emit another spike. Following the absolute refractory period, the neural membrane potential will be set to a small negative value  $u_{\text{refrac}}$  to account for a relative refractory time.

All parameters for the model and numerical simulation are given in Table 6.1.

### 6.1.2 Simulation results and discussion

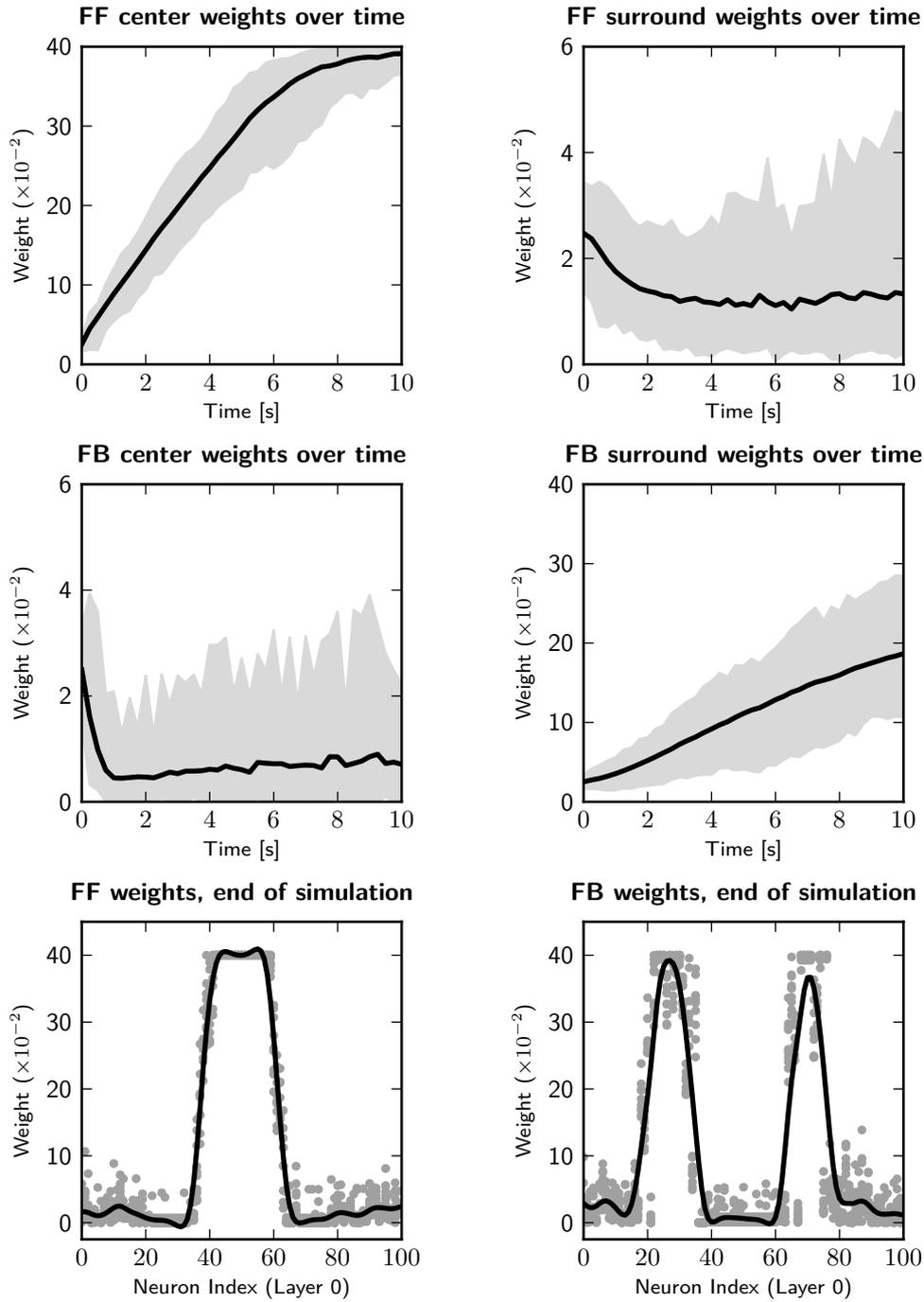
Receptive fields emerge after only short times of simulation. The evolution of weights is depicted in the top two rows of Figure 6.3. Examples of weights of the feed-forward and the recurrent feed-back connections at the end of a simulation are visualized in the bottom row of the figure. The figure shows that the weights within the central region around the stimulus increase in the forward kernel over time, whereas weights in the surround decrease. Thereby they form an on-center/off-surround receptive field. The figures also demonstrate that the inverse happens for weights in the recurrent weight kernels. Note the instability of the weights for longer simulation times due to unstable network dynamics.

The recurrent connectivity from layer 2 neurons drives neurons of layer 1 to their spiking threshold. In the simulation, neurons which received pre-synaptic activation from input neurons will be driven to this threshold. Note that the Heaviside function is not pivotal, but will act as a "short-term memory" effect for feed-forward input from the input layer. Only neurons which received input previously and thus have a positive membrane potential will get modulatory effects from recurrent collaterals. Furthermore, the system can be tuned by the parameter  $\theta_r$  which allows to set a minimum pre-activation threshold. Without this restriction, the recurrent activation is able to drive any neuron to the spiking threshold. An effect which is likely useful, but was considered irrelevant for the results presented here.

The network dynamics are clearly unstable. For long simulation times the network activity will saturate, i.e. all neurons will spike at their maximally possible frequency. However a more sophisticated model including homeostatic plasticity rules and sub-populations of inhibitory interneurons that modulate the firing activity of the excitatory neurons is likely able to counter-act chaotic dynamics. Such an elaborate model will be examined in future work.

## 6.2 Single neuron model of a grid cell

Recall that one of the primary goals of a MTS is to minimize the number of neurons required to encode transitions by bundling plausible transition points. A neuron should therefore sample from as many input locations as possible while subject to the *non-stationarity* and *coherency* constraints. Without these constraints, a single neuron would simply sample from each location in the input space to minimize the number of neurons. Taking the results from Section 6.1, a single neuron is thus assumed to express multiple on-center off-surround receptive fields. Pre-synaptic afferents and a single grid cell are modelled as follows.



### 6.2.1 On dendritic tree computation and the error function of a single transition neuron

In the presented model, a grid cell is assumed to expose several possible dendritic branches (from now on simply called *dendrites*) containing multiple potential dendritic spines. Thereby, a grid cell is able to sample from multiple locations of the input space.

The input space is represented by pre-synaptic neural activity which is arranged on a rectangular grid. Each dendrite therefore has a probability proportional to a weight  $\mathbf{w}_i := (w_i^1, \dots, w_i^N)$  to sample from the input space coordinate  $\mathbf{x}_i := (x_i^1, \dots, x_i^N)$ , where  $N$  corresponds to the number of dendrites. In other words  $N$  dendrites cover the whole input space, and the possible range of dendritic weights is given by  $\mathbf{w}_i \in [0, 1], \forall i$ . Note that the pre-synaptic input may be provided by  $M$  neurons with  $M \neq N$ , however this case is omitted in the analysis and simulation results.

It is further assumed that grid cells grow their dendritic trees towards pre-synaptic neurons due to a gradient process and thereby dendrites co-locate approximately uniformly with their inputs. Thus, the dendrites initially connect to all pre-synaptic neurons uniformly and thus uniformly cover the input space. Furthermore, multiple dendrites of a single neuron are assumed to overlap with the same pre-synaptic neuron. Thereby multiple dendrites sample from a single pre-synaptic input and hence will become co-active for a singular value  $\mathbf{x}$  of the input space. The extend of overlap, and thus receptive field size, is characterized by the parameter  $\sigma_1$ .

Instead of modelling the receptive field extents of pre-synaptic neurons in form of a continuous metric the following discrete wavelet-like function is employed. The receptive field kernels  $\rho^+$  and  $\rho^-$  for the on- and off-areas of a dendrite are modelled using the Minkowski distance. Here,  $p = 2$  is used, which reduces the Minkowski to the Euclidean distance. The kernels are subsequently binarized such that  $\rho^+, \rho^- \in \{0, 1\}$ . The binarization is rooted in the observation that neural spikes appear to be binary events and are only post-synaptically weighted due to synaptic efficacy. The kernels

---

**Figure 6.3 (previous page) – Evolution of weights in feed-forward (FF) and recurrent feedback (FB) projections and examples at end of simulation.** The top row shows the evolution of weights for the feed-forward connection from layer 0 to layer 1 and the middle row the evolution of weights of the recurrent feedback connections from layer 1 to layer 0. Black lines indicate mean values filtered with a Butterworth filter of order 2 and cutoff frequency 0.01, the gray area indicates maximal and minimal weights. The depicted weights are collected from 25 independent simulations with a simulation time of 10 s. The weights were subsampled every 250 ms. Note the difference of the axis values for center and surround weights. Examples for the weights at the end of the simulation are depicted in the bottom row. Each gray circle indicates the weight of a synapse. The black line are the weights filtered with a Butterworth filter of order 2 with a cutoff frequency of 0.20. The left column shows the weight distribution on the forward connections from layer 0 to layer 1, whereas the right column displays recurrent weights from layer 1 to layer 0.

## 6. A neural model of self-organizing grid cells

are formally defined as

$$\rho^+(\mathbf{x}, \mathbf{y}) = \begin{cases} 1, & \text{for } 0 \leq (\sum_{i=1}^N |x_i - y_j|^p)^{1/p} < \sigma_1, \\ 0, & \text{otherwise} \end{cases} \quad (6.3)$$

$$\rho^-(\mathbf{x}, \mathbf{y}) = \begin{cases} 1, & \text{for } \sigma_1 \leq (\sum_{i=1}^N |x_i - y_j|^p)^{1/p} < \sigma_2, \\ 0, & \text{otherwise} \end{cases} \quad (6.4)$$

where  $\sigma_2 = 2\sigma_1$  reflect the on-center and off-surround portions of the receptive fields. The receptive fields defined by  $\sigma_1$  and  $\sigma_2$  are abstractions of the results presented in Section 6.1. Note that indices are dropped if they are clear from context, e.g.  $\mathbf{x}_i := (x_i^1, \dots, x_i^N)$  reduces to  $\mathbf{x} := (x_1, \dots, x_N)$  as in the definitions above. The parameters  $\sigma_1$  and  $\sigma_2$  are related to the concept of *eigenresolution*, described in Section 7.2.

Given a weight vector  $\mathbf{w} := (w_1, \dots, w_n)$  describing the current dendritic weight distribution of a neuron, the total error is formally expressed as

$$F(\mathbf{w}) = \lambda L(\mathbf{w}) + (1 - \lambda)(E^+(\mathbf{w}) + E^-(\mathbf{w})), \quad (6.5)$$

where  $L(\mathbf{w})$  is the error with respect to the *dendritic load* of the neuron and  $E^+(\mathbf{w})$  and  $E^-(\mathbf{w})$  are the errors with respect to the transition constraints. The parameter  $\lambda$  allows to adjust the importance of each objective.

The *dendritic load* of a neuron accounts for the ratio of the input space which is covered by the neuron. In other words, given a discrete number of dendrites, it indicates how many of the dendrites are associated with pre-synaptic input. To minimize the number of required neurons, as is objective in an MTS, a single neuron has to associate with as many inputs as possible. Certainly, this objective is achieved when all weights are maximal. The *dendritic load error* is modelled as the mean squared error according to

$$L(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^N (1 - w_i)^2. \quad (6.6)$$

The dendritic load of a neuron is potentially either in support or conflict with the constraints of transitions, i.e. correlating to input signals and decorrelating from target symbols. These constraints are captured by the on-center and off-surround receptive fields of each dendrite. Their normalized errors are given by

$$E^+(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^N A_{ij}^+ (w_i - w_j)^2 \quad (6.7)$$

$$E^-(\mathbf{w}) = 1 - \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^N A_{ij}^- (w_i - w_j)^2 \quad (6.8)$$

with the normalization terms  $A_{ij}^+$  and  $A_{ij}^-$  assumed constant over the course of all simulations. They are defined as

$$A_{ij}^+ := \rho^+(x_i, x_j) \frac{\sum_{l=1}^N \rho^+(x_i, x_l)}{\sum_{m=1}^N \sum_{n=1}^N \rho^+(x_m, x_n)}, \quad (6.9)$$

$$A_{ij}^- := \rho^-(x_i, x_j) \frac{\sum_{l=1}^N \rho^-(x_i, x_l)}{\sum_{m=1}^N \sum_{n=1}^N \rho^-(x_m, x_n)}. \quad (6.10)$$

Due to the normalization terms, only unfavorable weight differences are taken into account in the computation of the error. For instance, a neuron which is associated with a pre-synaptic neuron that falls into its off-surround receptive field is penalized and expresses an increased error. By definition, the normalization terms capture boundary conditions and return the relative contribution to an error of a dendritic weight even if their receptive fields do not fully overlap with the input space. This happens, for instance, at borders of the simulated pre-synaptic input which is set up as a square region. The normalization hence prevents erroneous values introduced to the evolution of weights.

It is now possible to state the weight update for any weight  $w_k$  using the error function  $F(\mathbf{w})$ , namely

$$\frac{\partial}{\partial w_k} F(\mathbf{w}) = \frac{\partial}{\partial w_k} L(\mathbf{w}) + \frac{\partial}{\partial w_k} E^+(\mathbf{w}) + \frac{\partial}{\partial w_k} E^-(\mathbf{w}) \quad (6.11)$$

$$\begin{aligned} &= -\frac{2}{N}(1 - w_k) \\ &\quad + \frac{1}{N} \left( 4w_k - 4 \sum_i w_i A_{ki}^+ \right) \\ &\quad - \frac{1}{N} \left( 4w_k - 4 \sum_i w_i A_{ki}^- \right) \end{aligned} \quad (6.12)$$

$$= -\frac{2}{N}(1 - w_k) - \frac{4}{N} \left( \sum_i w_i A_{ki}^+ + \sum_i w_i A_{ki}^- \right), \quad (6.13)$$

which corresponds to a discrete convolution of the weights with the distance functions  $\rho^+$  and  $\rho^-$ , corrected by the weight terms  $A_{ij}^+$  and  $A_{ij}^-$ , respectively. The derivation of the individual error terms  $\partial/\partial w_k L(\mathbf{w})$ ,  $\partial/\partial w_k E^+(\mathbf{w})$ , and  $\partial/\partial w_k E^-(\mathbf{w})$  can be found in Appendix A.

It is assumed that synaptic strength in biological systems is limited and updates happen non-linearly. The weight change is therefore modelled using the Tanges Hyperbolicus. Hence, the discrete-time update during simulations follows according to

$$\mathbf{w}_{t+1} = \tanh(\mathbf{w}_t + \nabla \mathbf{w}_t). \quad (6.14)$$

where the dendritic weight update for gradient descend to minimize the error function  $F(\mathbf{w})$  follows immediately from

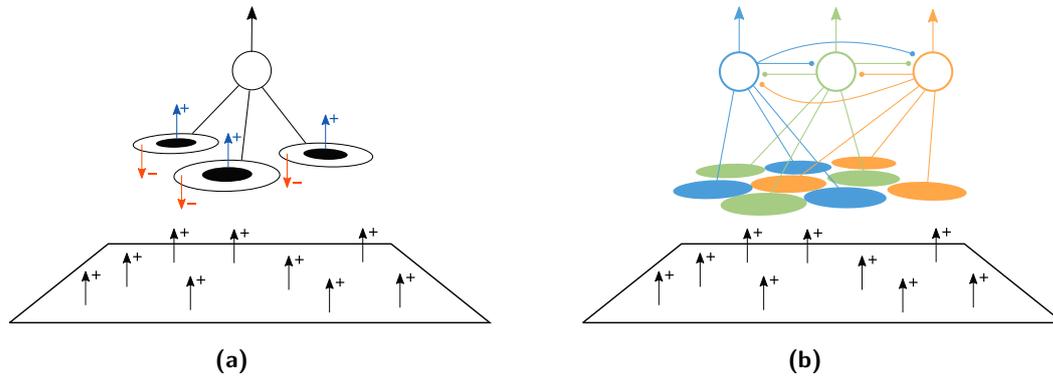
$$\nabla \mathbf{w}_t = -\eta \nabla F(\mathbf{w}_t), \quad (6.15)$$

using a constant learning rate  $\eta$ . Finally, weights are limited from below such that they cannot become negative, i.e.  $w_i \geq 0, \forall i$ .

The model is illustrated in Figure 6.4a. The figure depicts a single grid cell with three associated dendrites, each expressing an on-center off-surround receptive field. To minimize the error function, the on-center regions have to associate with afferents, whereas the off-regions need to decorrelate. However, on- and off-regions of different receptive fields are not allowed to overlap.

Note that the presented results apply to the continuous case in which, when using the Ricker (or Mexican-Hat) wavelet to model the receptive fields, the weight update

## 6. A neural model of self-organizing grid cells



**Figure 6.4 – Overviews of the single- and multi-cell models.** (a) A single cell has multiple dendritic branches, each expressing a center-surround receptive field. The cell associates to driving inputs from spatially modulated afferents, depicted by black arrows with a plus sign. However, it has to correlate only within center regions of its receptive fields, indicated by the blue arrows with a plus sign, while decorrelating in its surround regions, marked with orange arrows and a negative sign. (b) The inhibitory interactions between grid cells in a network of cells, indicated by lines with bullets at their endings, lead to a aligned arrangements of the receptive fields of each cell, depicted as filled circles. The objective of the entire network is to densely cover the spatially modulated afferents, depicted as black arrows.

is governed by the second derivative of the Gaussian kernel. The discretized version presented above allows to directly address boundary conditions and avoids expensive computations of exponentials during simulations, though.

### 6.2.2 Model characterization and simulation results

The emerging weight distribution as well as the numerical stability of the model were analyzed as follows. The input space, and therefore dendritic tree, was considered to be square and each dimension in the range  $[0, 1]$ . The weights of a grid cell were initialized to  $\tanh(1)$  with a chance of 10%, or 0 otherwise. The learning rate was kept static at  $\eta = 1.0$ . Several values for the parameters  $\lambda$  and  $\sigma$  were examined by simulating a single grid cell for 5000 iterations (arbitrary time units). The relative error importance  $\lambda$  was varied from 0.0 to 1.0 with a step-width of 0.1, whereas  $\sigma$  was varied from 0.05 to 0.20 in increments of 0.01. While varying one parameter, the other parameter was kept fixed. Each configuration of the parameters was evaluated in 40 simulations with  $N = 48^2$  dendritic weights and an additional 40 simulations with  $N = 64^2$  dendritic weights.

Gridness scores were computed using the common technique described by Sargolini et al. [313] (see also Appendix B for a brief overview). However, they were computed on the non-smoothed dendritic weights and not via intermediate generation of spike plots. The reason is that dendrites are assumed to sample from spatially modulated inputs and, therefore, average spiking behavior of the neuron precisely follows the distribution of dendritic weights. Spike generation, for instance using a Poisson process, based on the distribution of weights is thus considered an intermediate but unnecessary step.

Results for the numerical analysis of different receptive field size are depicted

in Figure 6.5. The figure shows results for  $\lambda = 0.50$  in the top and for  $\lambda = 0.65$  in the bottom row. Each row of the figure contain box plots for both  $N = 48$  as well as  $N = 64$  dendritic weights. Other configurations of  $\lambda$  in combination with several other values for  $\sigma_1$  were evaluated, but no qualitatively significant differences between the results were detectable. In all cases, the gridness of the weight distribution breaks down when  $\sigma_1 \geq 0.13$  and recovers for larger sizes. Furthermore,  $\sigma_1 \leq 0.07$  leads to impoverished gridness scores. As will be discussed below, the origin of the behavior is because of numerical constraints and not because of a principle issue in the model.

The impact of the relative error importance  $\lambda$ , given  $\sigma_1 = 0.10$ , is depicted in the top row of Figure 6.6. The results presented the bottom row of the same figure show a peak of performance for  $\sigma_1 = 0.13$  at which the simulations yield the best gridness scores while maintaining several grid fields in the dendritic weight distributions. Hence, the impact of  $\lambda$  was further analyzed for this value of  $\sigma_1$ .

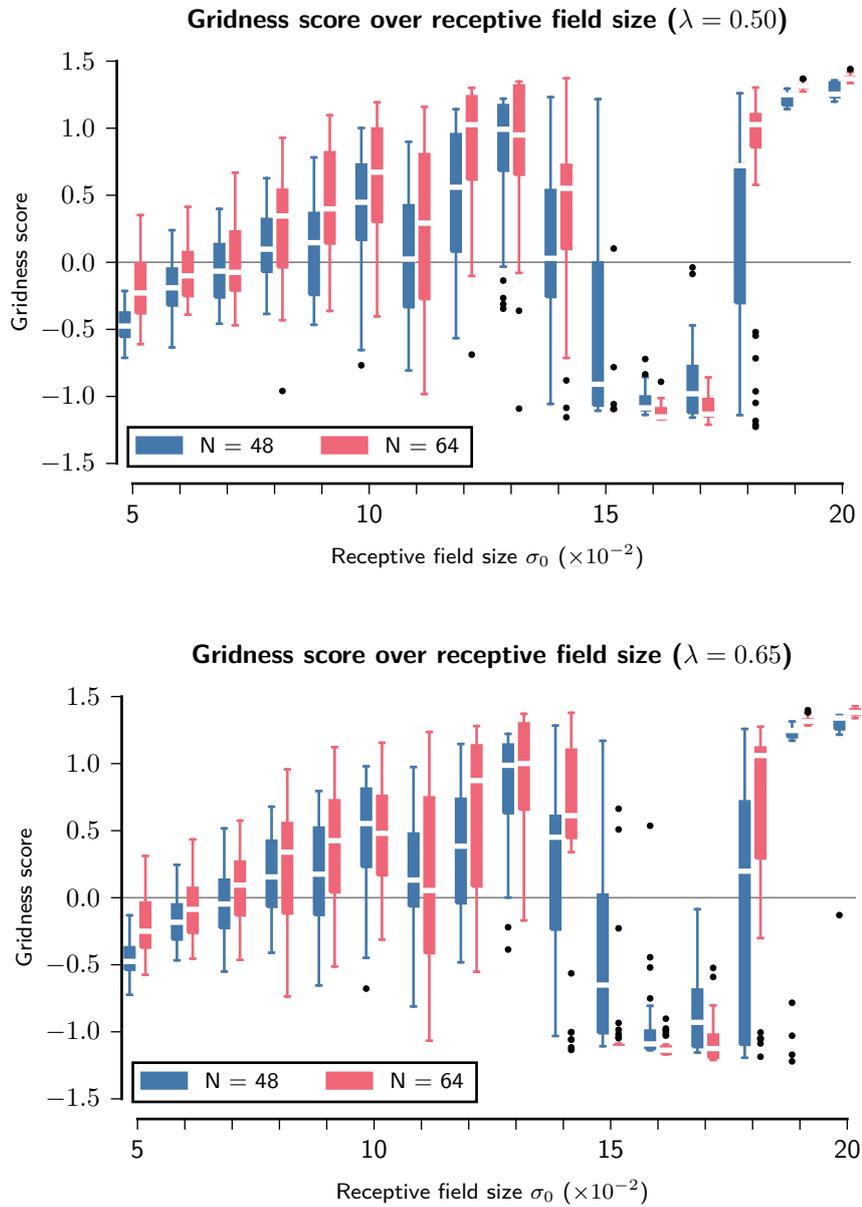
Average examples for the states of convergence of single cells with  $N = 48$  dendrites after 5000 iterations are depicted in Figure 6.7. The results for  $N = 64$  are qualitatively identical to  $N = 48$  and therefore omitted. The distance between circular weight fields corresponds to  $\sigma_1$  because  $\sigma_2 = 2\sigma_1$ . Smaller  $\sigma_1$  lead to numerical issues as is visible in the Figure 6.7. For instance, the first tile of the figure contains response fields which are mostly non-circular. The densest arrangement of fields that can be achieved on the square input is non-hexagonal but quadrangular for larger  $\sigma_1$ , as is shown in tiles for  $\sigma_1 = 0.15$ ,  $\sigma = 0.16$ ,  $\sigma = 0.17$ , and  $\sigma_1 = 0.18$ .

### 6.2.3 Interpretation of the model and results

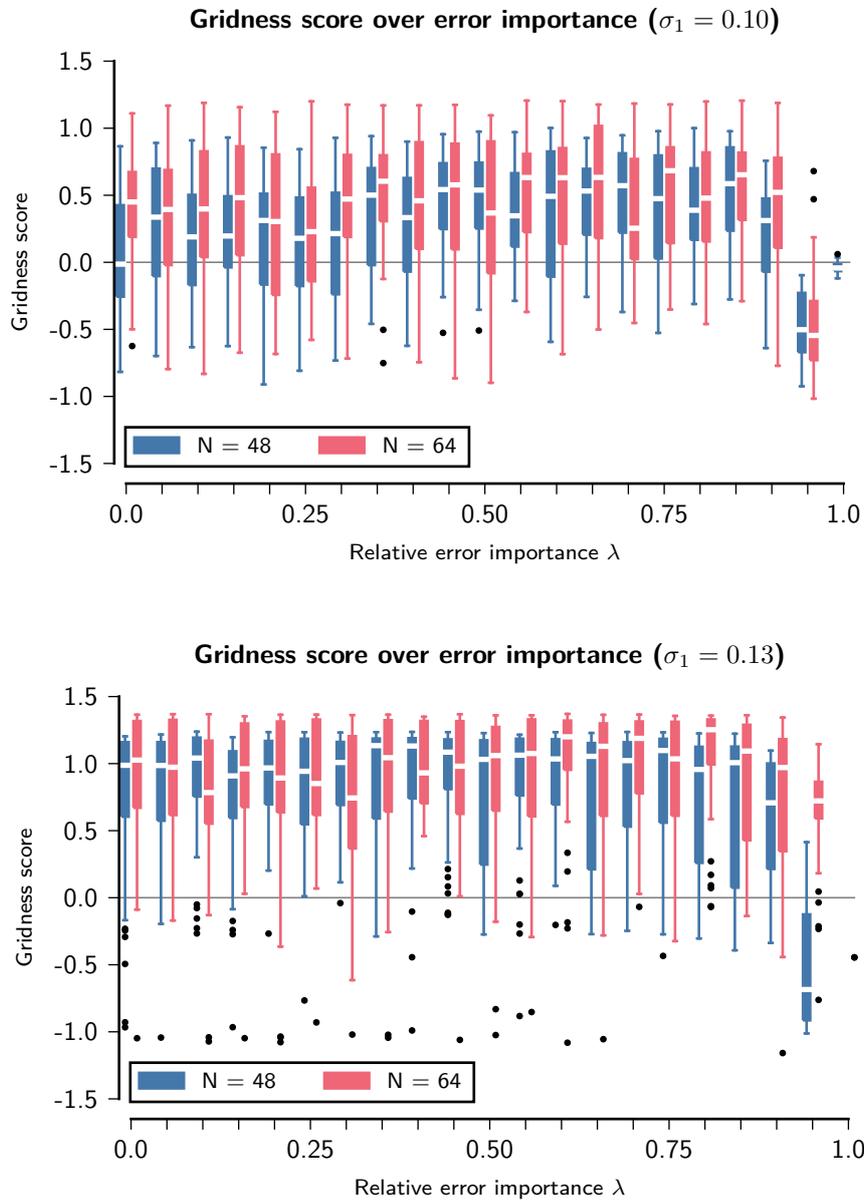
The coordinate  $\mathbf{x}_i$  directly corresponds to a location in two dimensional space due to considerations of simplicity of the simulations. However it is postulated that, on the one hand,  $\mathbf{x}_i$  could correspond to co-activity of spatially modulated neurons in a real neural network which are not necessarily place cells. Several boundary vector cells are likely candidates to provide input which yields unique spatial identification [13]. On the other hand,  $\mathbf{x}_i$  could correspond to a normalized physical location of a dendritic branch.

The weight update given in Equation (6.13) forms a Reaction-Diffusion System (RDS), known to be capable of producing Turing patterns [71]. Similar functions were used in CAN models of grid cells to form hexagonal firing fields [39, 70]. The major difference of these models to the model presented here is that the equation is not directly applied to the neural recurrent dynamics, i.e. the activity of neurons in a network, but to the excitability of a neuron based on its dendritic weights. Furthermore, these models typically include constants to tune the network dynamics and are susceptible to erroneous settings, whereas constants required in the equations above are a direct consequence of the error function given in Equation (6.5).

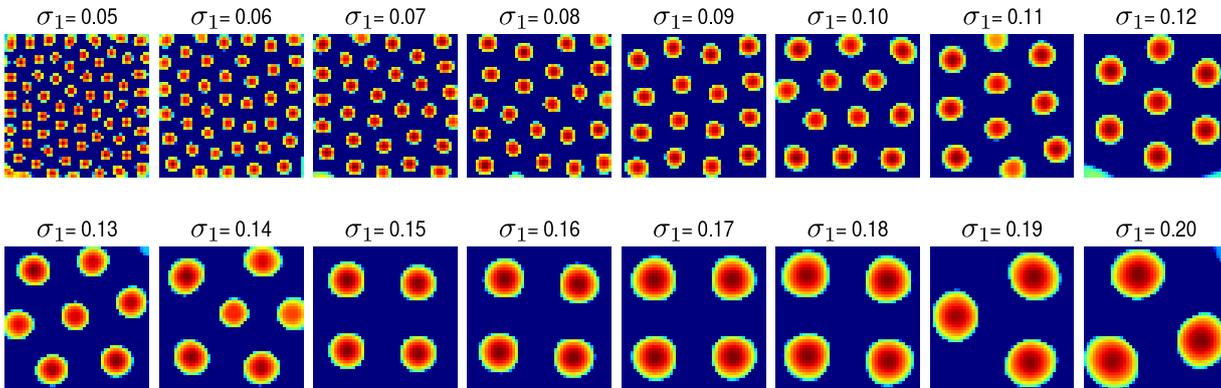
The results show a numerical problem of the simulation, observable in Figure 6.5. As soon as the receptive field size becomes too small, i.e.  $\sigma_1 \leq 0.07$ , the receptive field deteriorates and is not circular anymore due to the discretized square input bins. This can also be seen in the first tile of Figure 6.7. Thus, it is impossible for the dynamics to form a hexagonally dense arrangement of circular fields. A related issue is visible in the data shown for  $0.15 \leq \sigma_1 \leq 0.18$ . Here, the weight distribution forms one or two main blobs which are not detectable as a grid-like arrangement. The remarkable results for  $\sigma_1 \geq 0.19$  are only because a singular response field is



**Figure 6.5 – Single grid cell model, gridness scores over receptive field sizes.** Gridness scores were computed directly on the emerging weight maps after 5000 iterations. The relative error importance was set to  $\lambda = 0.50$  in all simulations presented in the top row. The bottom row contains results for  $\lambda = 0.65$ . Weights were initialized with a chance of 10% to  $\tanh(1)$  and 0 otherwise. Each depicted configuration was simulated for 40 times.



**Figure 6.6 – Single grid cell model, gridness scores over error importances.** Gridness scores were computed directly on the emerging weight maps after 5000 iterations. The receptive field size was set to  $\sigma_1 = 0.10$  in all simulations of the top row, and to  $\sigma_1 = 0.13$  in the bottom row. Weights were initialized with a chance of 10% to  $\tanh(1)$  and 0 otherwise. Each depicted configuration was simulated for 40 times.

Weight maps of single neurons with  $N=48$  dendritic weights after 5000 iterations

**Figure 6.7 – Single grid cell model, state of convergence after 5000 iterations for varying  $\sigma_1$ .** The simulated cell had  $N = 48$  dendrites and  $\lambda = 0.50$ . Large values of  $\sigma_1$  show problems of the simplified rectangular receptive field. Results for  $N = 64$  are qualitatively identical.

formed, which leads to perfectly hexagonal arrangements in the auto-correlogram used to compute the gridness score. Furthermore, the small number of dendritic weights, which was chosen due to reduce simulation times to an acceptable duration, introduce small but mostly negligible problems. Nevertheless, the results depicted in Figure 6.6 and Figure 6.5 demonstrate that the model is able to form stable hexagonal fields in most of the cases despite these numerical inaccuracies.

The square pre-synaptic input space is biologically unlikely. However, spatially modulated neurons with response fields localized only in single or few locations have been observed, e.g. in form of place cells [265]. Furthermore, the boundary vector space in combination with head direction information allows to represent arbitrary locations unambiguously in a square environment [13]. Thus, spatially modulated neurons which are confined to the square experimental environment are not unlikely, thereby providing a limited input-space which is also quadratic.

It is proposed that one of the primary elements of the input space to grid cells is boundary vector information. Such a space, anchored egocentrically, provides information about distances to geometrical boundaries and appears to be an ideal candidate to generate hexagonal fields. Boundary information was used in a model for place cell formation [13], whereby it was shown that boundary vector cells provide sufficient information to identify locations. The latter is crucial in the sampling process for the dendritic tree, presented here. Furthermore it was reported that grid fields arrange due to the geometry of an environment [197, 199], clearly in favor of the proposition.

In the model, a single grid cell associates with multiple input state representations. Thus, it is proposed that local dendritic computation in grid cells is more involved than what is suggested in most other models for grid cells, with the notable exception of the model by Kerdels et al. [178]. It is likely that novel techniques, such as protein calcium imaging [57], will be able to determine if dendritic computations are indeed performed by grid cells, or if individual dendritic spines of grid cells perform distributed computations. Specificity towards individual inputs and distributed computation were already reported for other cells on the level of dendritic branches,

spines, and even synapses [30, 63, 237, 320]. Furthermore, it has been reported that dendritic spines expose cooperation based on their physical location [374].

The growth process for the overlap of branches, and therefore receptive field sizes, could be either governed genetically, or modulated due to feedback from other systems. In case of a genetic primer, a simple rule or gradient based mechanism to govern the sub-division or novel growth of branches is expected. An elaborate simulation based on fractal rewriting systems, for instance, L-systems, could provide realistic dendritic trees based on such simple rules which, in turn, could then be used to study the formation of different grid field sizes. L-systems were already successfully used to model the growth of neurons and their dendritic spines [399], and fractal geometry in general appears to be a fundamental concept in biological systems [159]. It is expected that in such a rewriting system, a simple rule in which a dendritic branch is *rewritten* by two or more smaller branches would not only allow the generation of smaller receptive fields, but already provide a genetic pointer for the generation of discrete receptive field sizes. The latter case of receptive field tuning, i.e. driven by recurrent activity, would require ongoing activity-dependent structural plasticity of neurons. This form of plasticity was indeed observed and attributed to calcium signalling [387]. A potential external signal to modulate, general principle behind, and behavioral necessity for discrete scales of receptive grid field sizes in light of MTT are presented and discussed in Section 7.2.

So far, the model assumed an input space of a certain size which depends on the number of dendrites specified during the simulation. Certainly, a non-artificial neuron is also limited in the number of synapses that it can form. Thus, the possible associations of one neuron to input patterns is also restricted and allows to predict expected numbers of cells given an input space. This calculation is postponed to Section 7.2, though.

### 6.3 Competitive network model of grid cells

The single cell model presented in the previous section has a limitation with respect to biological plausibility. It assumes that all inputs are presented uniformly to the grid cell and that it learns an optimal representation to minimize the average error due to an *offline learning* strategy. However, it is unlikely that the real grid cell network is presented with inputs in such a manner that the self-organizing process can pre-compute the optimal weight distribution. It is equally unlikely that grid cells are not subject to plasticity effects which alter these weights over time. In contrast, it is more likely that an online process adapts grid cells which express continued plasticity. Furthermore, a single grid cell cannot cover the entire input space. Therefore, multiple cells are required to jointly associate with *all* possible locations in the environment for the purpose of storing all feasible transitions. Hence, each cell has to adapt due to its local internal representation error introduced in the previous section, as well as a global error signalling the *coverage* of the input space.

Consequently, an online learning process for the minimal number of three grid cells is presented for a moving agent in this section to address these shortcomings. The process and weight updates are based on the single cell model, but adapted such that the weight updates return stable results with only singular update events of the current location of the animal. Additionally, an error term with respect to the

co-activity of grid cells is introduced. The weight updates are then performed using a competitive winner-take-all strategy. The section closes with a discussion of the model, its results, and its biological plausibility.

### 6.3.1 Model description and network dynamics

The network consists of  $N_g$  neurons, with  $N_d$  dendritic weights  $w_i$  each. Each dendrite is tuned to sample from pre-synaptic inputs which are spatially modulated for a location  $\mathbf{x}_i$  as described above. Likewise, the on-center/off-surround receptive fields are tuned by the parameters  $\sigma_1$  and  $\sigma_2$ .

The temporal dynamics of pre-synaptic inputs and dendritic activation are modelled as follows. Any pre-synaptic activity for a location  $\mathbf{x}_t$  at time-step  $t$  of the simulation stimulates a total of  $D_t = B_t \cup C_t$  dendrites of a single cell.  $B_t$  and  $C_t$  are the dendrites with an overlap of either their on- or off-portions of their receptive fields and the input stimulus. The results of Section 6.1 showed that the weights in the off-surround vanish, though. Hence, only the activation of  $B_t$  is considered to drive a cell to its spiking threshold. Conversely, decorrelation is modelled only for dendrites  $C_t$ .

The neural activity is computed by evaluating  $B_t$  without considering explicit spiking behavior. Furthermore, the activity collected by the  $B_t$  dendrites is used for a winner-take-all selection of the best matching neuron. Given the set of stimulated dendrites  $B_t$ , grid cell activity  $a_n(B_t)$  for each grid cell  $n \in 1, \dots, N_g$  is computed as the weighted sum over all dendrites which receive stimulation. Formally,

$$a_n(B_t) = \frac{\sum_i^{N_d} w_i}{|B_t|}, \quad (6.16)$$

where  $|B_t|$  corresponds to the number of dendrites receiving activity at time  $t$ . Pre-synaptic input activity is considered binary and modulated only locally in the post-synaptic neuron by the corresponding dendritic weights.

Recall the error function of the single cell model given in Equation (6.5). It contains independent terms for the dendritic load as well as the correlation/decorrelation constraints. In the competitive network model, all neurons have to jointly cover the input space but uniquely identify transitions. Therefore, co-activity of grid cells is penalized by introduction of the following additional non-linear error term.

$$C_n(B_t) = \frac{1}{2(N_g - 1)} \sum_{m \neq n} a_m(B_t) a_n^2(B_t) \quad (6.17)$$

Consequently, the entire error function for neuron  $n$  at time-step  $t$  during the online learning procedure is given by

$$F_n(\mathbf{w}, B_t) = L_n(\mathbf{w}) + E_n^+(\mathbf{w}) + E_n^-(\mathbf{w}) + C_n(B_t), \quad (6.18)$$

with  $C_n(B_t)$  as stated above and  $L_n(\mathbf{w})$  given by

$$L_n(\mathbf{w}) = \frac{-1}{N_g N_d} \sum_{i=1}^{N_d} (1 - w_i)^2. \quad (6.19)$$

The error terms  $E_n^+$  and  $E_n^-$  are subject to the winner-take-all mechanism. At each time step  $t$ , the winner-take-all mechanisms selects the most active grid cell according to  $a_n(B_t)$ , and only the winner receives error signals for  $E_n^+$  and  $E_n^-$ . In other words, the winner-take-all mechanism allows only the winner to associate with the presented input and prevents all other neurons from updating their correlation-decorrelation objective.

The error function leads to the following online update rule for weight  $w_k$  of a neuron  $n$  and time step  $t$ .

$$\begin{aligned} \frac{\partial}{\partial w_{n,k}} F(\mathbf{w}_n, B_t, C_t) = & -\frac{2}{N_g N_d} (1 - w_{n,k}) + \frac{1}{N_g - 1} \sum_{m \neq n} a_m(B_t) a_n(B_t) \\ & + \frac{\partial}{\partial w_{n,k}} E^+(\mathbf{w}_n, B_t) + \frac{\partial}{\partial w_{n,k}} E^-(\mathbf{w}_n, B_t, C_t), \end{aligned} \quad (6.20)$$

where

$$\frac{\partial}{\partial w_{n,k}} E^+(\mathbf{w}_n, B_t) = \begin{cases} \frac{-4(N_g-1)}{N_g N_d} |B_t| w_k, & \text{if } n = \arg \max_m a_m(B_t) \\ 0, & \text{otherwise} \end{cases} \quad (6.21)$$

and

$$\frac{\partial}{\partial w_{n,k}} E^-(\mathbf{w}_n, B_t, C_t) = \begin{cases} \frac{4(N_g-1)}{N_g N_d} |C_t| w_k, & \text{if } n = \arg \max_m a_m(B_t) \\ 0, & \text{otherwise} \end{cases}. \quad (6.22)$$

The weight update of weight  $w_{n,k}$  in discrete time steps follows accordingly.

$$w_{n,k,t+1} = \tanh \left( w_{n,k,t} - \begin{cases} \frac{1}{3} \eta(t) \frac{\partial}{\partial w_{n,k}} F(\mathbf{w}_n, B_t, C_t), & \text{if } w_{n,k} \in D_t \\ 0, & \text{otherwise} \end{cases} \right) \quad (6.23)$$

Note that thereby only dendritic weights which are either in  $B_t$  or  $C_t$  are updated. Any dendrite which is not subject to pre-synaptic stimulation remains at its previous weight. As in the single cell model presented in Section 6.2, weights are subsequently clamped from below such that  $w_{n,k} \geq 0$  for any neuron  $n$  and weight  $k$ .

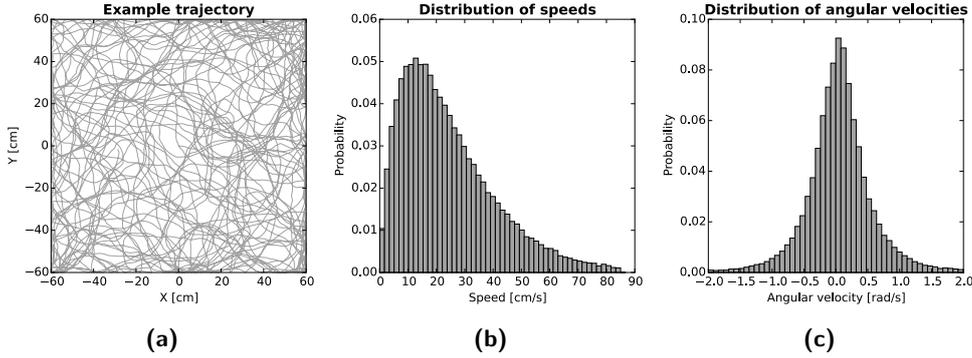
The learning rate  $\eta(t)$  is self-tuning and defined according to

$$\eta(t) = \exp \left( -\frac{1}{s_{\text{avg}}} s_t^2 \right), \quad (6.24)$$

where  $s_t$  is the animal's speed and  $s_{\text{avg}}$  is the average expected running speed. Although  $s_{\text{avg}}$  was pre-computed for the presented simulations, a running average yields the same results.

The entire network is illustrated in Figure 6.4b. The cells interact using only inhibitory interactions, drawn as lines with circular endings. To minimize the objective function, the cells will arrange their receptive fields accordingly and form a grid module.

## 6. A neural model of self-organizing grid cells



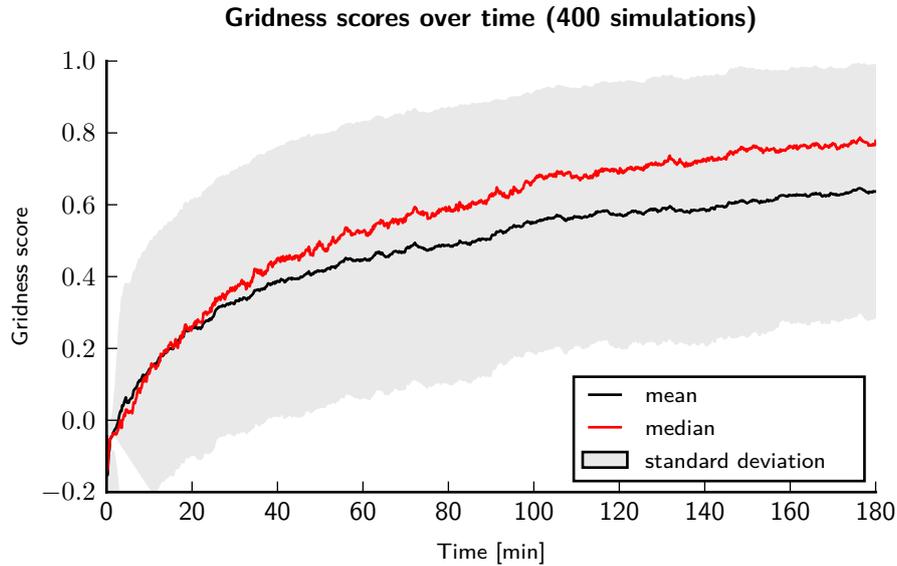
**Figure 6.8 – Trajectory example and movement statistics.** (a) One example of a trajectory used during simulations involving random exploration of a square environment. (b) The distribution of speeds closely resembles data reported from real rodents. (c) The angular velocities are tuned according to real data as well.

### 6.3.2 Methods and simulation results

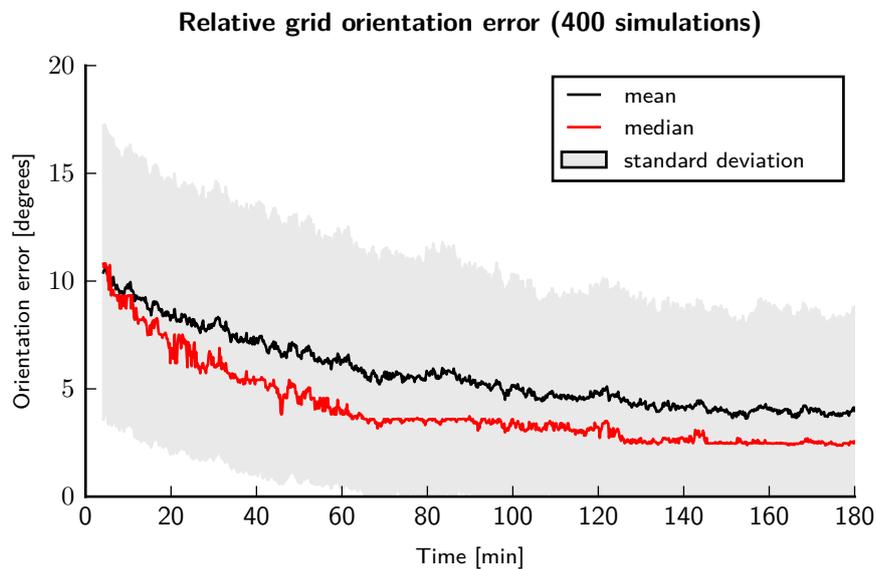
Virtual input trajectories with movement statistics close to real data of rodents were used. A novel trajectory was generated for each simulation. An example of a trajectory as well as statistics on angular velocity and running speed are depicted in Figure 6.8. The location of the animated animal was presented to the network every 10 ms. The duration of one training session was 3 h to observe long term effects. Furthermore,  $N_g = 3$  and  $N_d = 48$ . The weights of each neuron were initialized such that they had a 10% chance to be set to 1, or were set to 0 otherwise. The receptive field size was set with parameters  $\sigma_1 = 0.10$  and  $\sigma_2 = 2\sigma_1$ . The network was simulated for a total of 400 times. Note that  $\lambda$  was dropped from the equations due to the results of the previous section. The receptive field sizes were determined by  $\sigma_1 = 0.10$  and  $\sigma_2 = 2\sigma_1$ .

As discussed in Subsection 6.2.2, the gridness scores were computed directly on the dendritic weights and not on intermediate spike response plots. Certainly, the ongoing plasticity of the weight distribution may introduce changes in the location of formed grid fields. However, it was observed that once the fields formed, they remained stable throughout the rest of the simulations except for subtle re-arrangement of the fields. The increase of the gridness score over time as depicted in Figure 6.9 and the stability of the fields can be observed in the examples presented in Figure 6.11. This means that although the cells were subject to persistent plasticity, the dendritic weights stayed at their peak locations once the network formed pronounced grid fields. Furthermore and although the cells were subject to continued competition within the network, the fields only moved towards an improved packing of fields. This is also expressed in the steady increase of the gridness score over time.

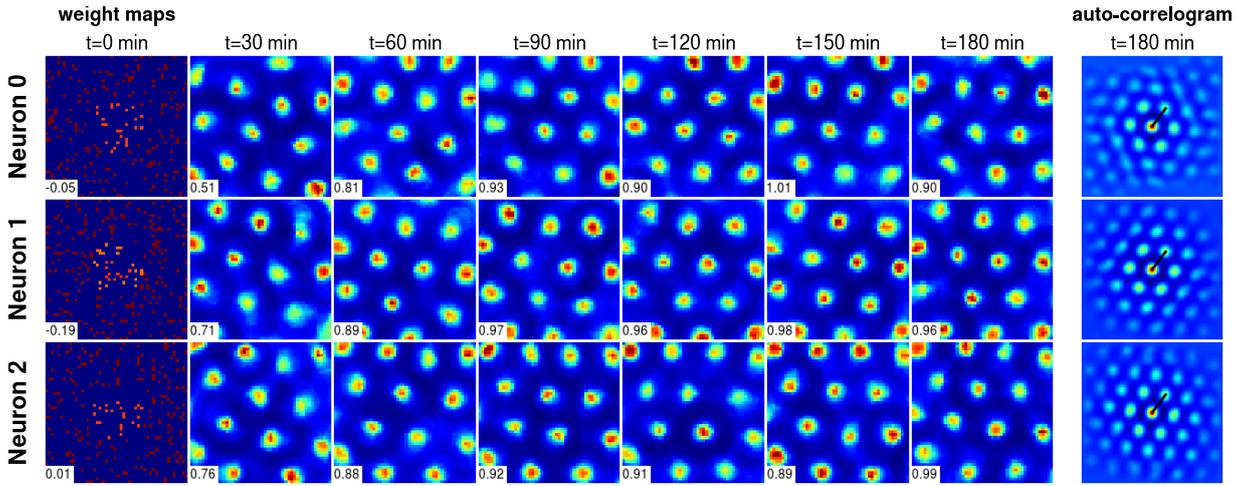
In addition to the gridness score, relative orientation errors between the cells of each simulation were computed. Thereby it is possible to assess if the grid cells are aligned or form random alignments to each other. Likewise the characterization by Hafting et al. [135], the alignment of the dendritic weights of each grid cell was assumed to be in the range from 0 to 60 degrees. Subsequently, the relative orientation between cells was computed which gives results in the range from 0 to 30 degrees. Finally, the average error of each simulation time-step was computed. The calculation



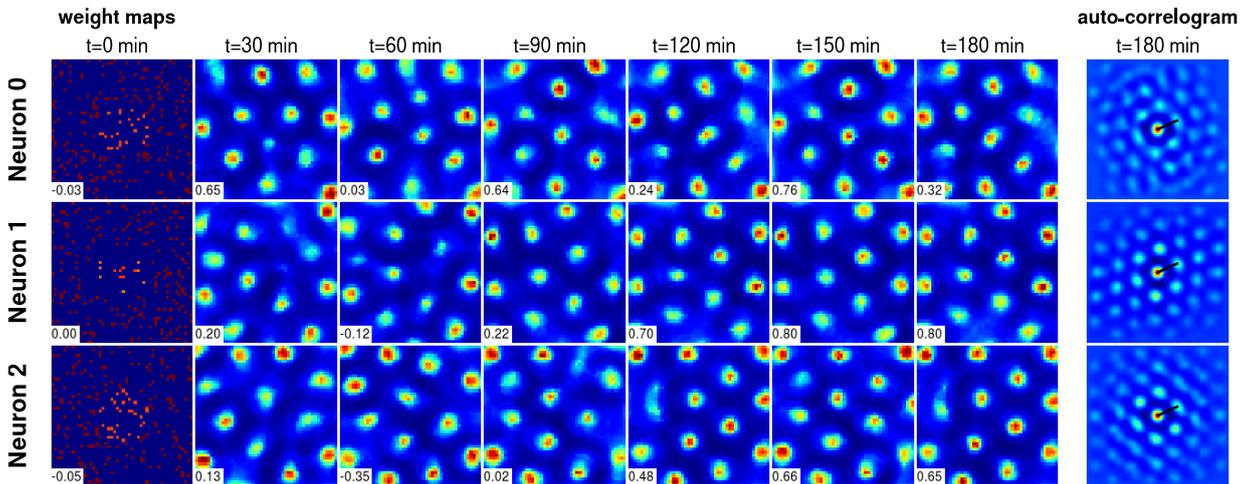
**Figure 6.9 – Competitive grid cell network, gridness score over time (400 simulations).** The data was computed using 400 simulations of a network of  $N_g = 3$  cells with competitive dynamics,  $N_d = 48$  dendritic weights each, and  $\sigma_1 = 0.10$ . The median and mean of the gridness scores stay above zero after about 2.5 or 4 minutes, respectively. The standard deviation of the gridness scores is comparably large, but may be due to numerical issues introduced by setting values for small  $N_g$  and  $\sigma_1$ .



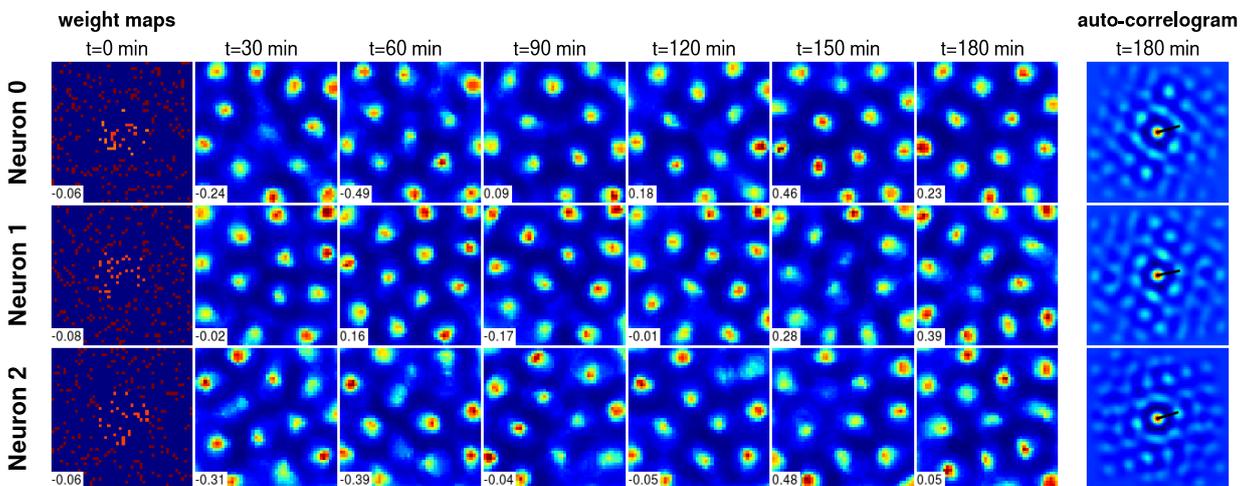
**Figure 6.10 – Competitive grid cell network, relative orientation error over time (400 simulations).** The first four minutes of data are cut off because an orientation could not be computed with certainty. Over time, the median and mean of the orientation error approach zero but stay at an approximately 2.5 degrees offset. So far it is unclear if the offset is introduced by the network dynamics or numerical resolution of the auto-correlogram which was used to compute the orientation.



Simulation #311. Final average gridness score = 0.949. Final relative orientation error = 0.000°.



Simulation #60. Final average gridness score = 0.589. Final relative orientation error = 0.000°.



Simulation #19. Final average gridness score = 0.223. Final relative orientation error = 3.339°.

of the orientation error is described in detail in Appendix B. Orientation errors over time for all 400 simulations are depicted in Figure 6.10.

Wall-offset orientations were automatically computed for all neurons using their primary orientation extracted from their auto-correlograms. So far, a clear preference for a specific value was not observable in the data. Some cells showed an alignment of their weight fields in perfect alignment of the walls. Yet, most of the cells settled for an orientation offset in the range from 5 to 12 degrees. However, results from visual inspection and manual analysis for many simulations indicate that the weight fields are subject to skewing and shearing effects near walls. Consequently, the orientation of the fields along walls is slightly different to the orientation of fields in more central areas of the arena. Examples for the skewing and wall-offset are observable in Figure 6.11. In all simulations the response fields at the end of the simulations (time-step  $t = 180$  min) appear to be slightly curved. The effect is especially prominent in simulation 311 (top row), neuron 0 and in simulation 60 (middle row), neuron 2. However, further studies are required to characterize the effects and investigate if the model indeed generates wall-offsets comparable to the findings presented by Stensola et al. [337].

The average gridness score was above zero at the end of the simulations in all simulations, and only in 11% of the simulations one single cells had gridness score below zero. In addition, the mean gridness score reached a gridness of 0.0 after only about 4.0 min of simulated time, and the median was permanently above zero already after approximately 2.5 min. The median and mean gridness scores as well as the standard deviation for all 400 simulations are depicted in Figure 6.9. The standard deviation appears to be quite large but may be due to numerical issues discussed previously and introduced by the small number of dendritic weights as well as the receptive field size  $\sigma_1$ .

The evolution of dendritic weights of exemplary simulations with high, average, and low gridness scores are depicted in Figure 6.11.

### 6.3.3 Discussion of the model and its results, predictions, and future work

The angular error between neurons decreases over time, depicted in Figure 6.10. Therefore, the network minimizes alignment errors and produces neurons with shared

---

**Figure 6.11 (previous page) – Examples for the weight evolution and final auto-correlation maps.** Each block displays a single simulation and displays the dendritic weight maps with respect to spatial location of each of the three neurons at several time-steps. Furthermore, the auto-correlograms of the weight maps after the final time-step are given. The auto-correlograms include a black line from the center to the closest peak which was used for the computation of the orientation (for details see Appendix B). The weight maps at the first time step ( $t = 0$  min) are the maps *after* the first location was presented to the cells. Thus they express altered weights at the center of the map because the animal always started the exploration in the center of the arena. The first block contains a simulation with a final average gridness score of 0.949 and relative orientation error  $0.000^\circ$ , the second block has final score 0.589 and error  $0.000^\circ$ , the bottom block has score 0.223 and error  $3.339^\circ$ . The numbers in the white inlays in each weight map are the respective gridness score computed for the map.

## 6. A neural model of self-organizing grid cells

orientation. Thus, the competitive dynamics are able to generate modules of grid cells with shared configurations. Furthermore, the cells express fields with phase offsets such that the entire input domain is covered. As can be observed in Figure 6.9, the fields form very early and the gridness score continues to increase over the course of the simulations. Furthermore, the weight fields remain stable and move only with a coincident increase of the gridness score. Thus, the dynamics presented in Subsection 6.3.1 describe an optimization process which converges to hexagonal arrangement of the weight fields.

The model postulates that the velocity of an animal contributes to the formation of grid cells in form of a self-tuning learning rate and is based on the following argument. The certainty to be at a specific location should be inversely proportional to the running speed. For instance, an animal at rest is very certain about its current whereabouts. However, this certainty should decline with increased speed. As linear speed cells were found to exist in the mEC [195], speed information is indeed available. The model however makes a strong prediction about the interaction between speed cells and grid cells, as the learning rate of grid cells depends on the speed. This modulatory effect is expected to be either facilitated via inhibitory inter-neurons in such a manner that speed cells suppress currently active grid cells, or in a way whereby *future* grid cells are more strongly supported by speed cell activity than grid cells which are associated with temporally and spatially nearby locations. As the mEC shows almost only inhibitory recurrent activity [70], it is more likely that the modulation is facilitated using inhibitory feedback. Changing the impact of speed from a non-linear contribution as defined by Equation (6.24) to a constant value decreased the stability of grid fields in the model. This effect is therefore also expected to appear in real rodents.

The model is independent of heading direction. Rather, grid cell responses depend on the input space of spatially modulated pre-synaptic neurons. It is likely that this space is spanned by head direction cells in combination with boundary information. Thus, an elaborate model including intricate designs of the pre-synaptic neurons, including head direction cells, will likely yield grid cells that fire more strongly with respect to the head direction of the animal and less with the movement direction. Therefore it is expected that the results observed by Raudies et al. [293] can be explained in future studies and models which employ the necessary detail with respect to pre-synaptic neurons.

The results presented above allow abstractions of the computations and algorithms performed by grid cells. The hexagonal arrangement of grid cell firing fields can be modelled as the densest packing of circular or particle-like sampling regions of an input space. These particles need to interact in such a way that they are not overlapping, but still as tightly packed as possible with respect to their on-center and off-surround areas. In fact, it is postulated that optimal dense packing of particles with soft boundaries are the reason for shearing effects as well as wall-offsets as observed in biological data [337]. A computational model how multiple grid cells can co-ordinate their sampling regions such that grid cell responses are aligned was already suggested by Kerdels et al. [178]. In this study, grid cells are represented by a GNG and sample from their input space similar to the method presented here. Although the study focuses on spatial sampling and not on transitions, it shows that the realignment issue of grid modules can be understood in algorithmic terms.

An additional benefit of the abstraction of grid fields in form of elementary samplers is the possibility to study requirements for the pre-synaptic representation. In short, the pre-synaptic input space is required to present sufficient information for a disambiguation between places. One likely candidate for such an input space is examined in the preliminary results presented in Appendix C. Conclusively, it is likely that grid cells inherit their *metric* information and accuracy from pre-synaptic neurons and their corresponding sensory and representational resolution.

Pre-synaptic activity is required to be spatially modulated. It is therefore proposed that boundary information is one of the primary inputs for self-organizing grid cells. Boundary vectors have been successfully used in a model which describes place cell firing fields [13]. Consequently, they are likely candidates for spatial discrimination, which was assumed to be available as pre-synaptic input in the model presented here. Preliminary results indicate that the boundary vector space allows to form centralized, approximately hexagonal, sampling locations (see Appendix C).

Some abstractions which were used in the model limit its biological accuracy. For instance, the currently employed winner-take-all mechanisms in combination with the co-activation depression show only limited success to form hexagonal grid fields when more than three cells are simulated. The cells express localized response fields, but due to the non-graded absolute winner selection, only the winning neuron correlates with the input. Furthermore, the co-activation suppression strongly decorrelates the activity of neurons which are active at the same time. Though both mechanisms are biologically inspired, they are not plausible. A hard winner-take-all mechanism would require exceptionally fast recurrent inhibitory activity. And indeed, it was observed that the HF is governed by inhibitory collaterals which operate in the range of milliseconds [86]. It is also likely that this observation will be made for the mEC. Nevertheless, the small time window between feed-forward excitation and recurrent inhibition may be sufficient to let grid cells organize with overlapping fields. Thus, a model based on neurons with non-linear temporal dynamics, for instance a Leaky-Integrate and Fire (LIF) model in combination with STDP, is likely to be provide overlapping responses.

Future models and electrophysiological recordings have to investigate the receptive fields of grid cells more rigorously. The receptive fields of dendrites were assumed to be perfectly circular in the results presented here. Furthermore, neural activity is computed simply by summation of activity of pre-synaptic states and the corresponding dendritic weights. However, it is expected that the receptive fields of the dendrites of grid cells express complex interactions with the tuning curves of pre-synaptic neurons due to the individual tuning of each dendrite. Furthermore, pre-synaptic spike characteristics, for instance if the pre-synaptic neurons are bursting or not, may have an influence.

The results presented here are used in Chapter 7 to develop an abstract model of the interactions between grid and place cells. The model is used to examine computational consequences of the transition encoding which, in turn, leads to discrete scales and the proposition of a scale-space model of grid cells.



## Chapter 7

# Algorithmic exploration of the entorhinal-hippocampal loop

Chapter 5 formally introduced MTT. Furthermore, the optimal technique to store arbitrary transitions in a MTS was analyzed. Subsequently, optimal encoding was deduced for metric spaces. One of the main results was that, given an optimal sampling process for a continuous metric space, the minimal number of encoders to store transitions in an infinite space is finite. The ideal arrangement of encoders is hexagonal for the two-dimensional case. Using these theoretical observations, a biologically plausible model for grid cell firing was derived in Chapter 6. Simulations showed that a competitive network of grid cells arranges hexagonal firing fields as a result of dendritic tree computation, even when the input space is arranged sub-optimally, i.e. non-hexagonally. Now, these results are used to develop an algorithmic perspective for the interactions between place and grid cells.

First, a universal MTS  $\mathcal{M}$  is considered. In such a system, temporal transitions are learned during explorative or goal-directed navigation. The system is described algorithmically to introduce required concepts and abstractions. Then, spatial transitions are incorporated and the combined spatio-temporal system  $\mathcal{L}$  contrasted to the temporal setting. Subsequently, the spatio-temporal system is examined in the light of behaviorally significant issues. To solve these issues, a scale-space encoding of transitions is introduced. It is argued that, in the optimal case, the scale-space representation exhibits a scale increment of  $\sqrt{2}$  with respect to grid field radii and grid field periods for consecutive scales. As consequence of the computational properties, a novel functional hierarchy of the entorhinal-hippocampal loop is suggested. The terms *spatial symbol (neuron)* and *place cell*, and *spatial transition (neuron)* and *grid cell*, are used synonymously throughout the text.

During all these steps, biologically rigorous modelling is sacrificed in favor of abstract methods. It is believed that the algorithmic interactions between different areas which are responsible to learn and recall transitions for goal-directed spatial navigation can be expressed easier without involving non-linear neuro-dynamics. Therefore, the subsequent sections will use the following simplifications. Both grid and place cells are assumed to form independently on the basis of optimally sampled spatially modulated input, e.g. boundary information. Furthermore, it is assumed that the grid cell response formed already and is arranged hexagonally. Other potential afferents for the formation of place cells, for instance odors, will be ignored.

Certainly, the simplifications make this chapter more speculative with respect

to biological implementations. The abstractions introduce inaccuracies and issues, especially due to discretization. The chapter thus concludes with a detailed discussion about the biological plausibility of the proposed model, propositions about functional properties of certain neurons, and relations to observations made in real recordings.

## 7.1 The Universal Multi-Transition System as Growing Neural Gas

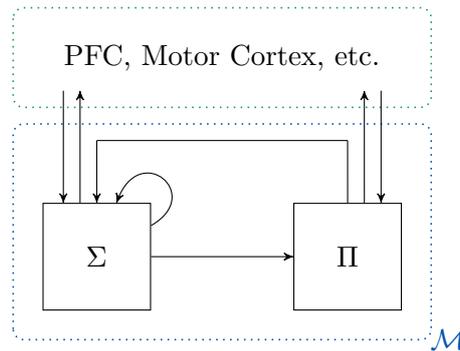
Cuperlier et al. previously presented a biologically plausible model of a temporal transition system [73, 74, 146]. Their system modelled several cells of the HF, for instance place and presumed transition cells, and was tested in a robotics scenario. While moving through an arena, the agent learned places, transitions, and the associated motor commands from one behaviorally relevant location to another [73, 74]. With the help of RL, rewards were propagated along the transitions from one location to another. Given sufficient exploration, the agent was able to compute multiple trajectories to target locations. Due to the reward accumulation by RL, the agent was also able to select the trajectory which maximized the reward [146]. Furthermore, the system was capable to recover from the *kidnapped robot* problem during which the agent is relocated to an arbitrary place. Afterwards, the robot had to either drive back to its previous location or continue with its previously assigned task. Both situations required that the robot was able to assess its location based on memory, and find suitable trajectories to the objective.

The algorithms presented in this and the following sections are reduced in scope due to the previous work by Cuperlier et al. [73, 74]. Namely, only the steps to determine if a trajectory from a start to a goal location exists are included. Furthermore, acquisition of novel transitions and places using involved cell types are incorporated. Selection of a winning trajectory, e.g. based on some reward signals, is left for future work. Despite this cutback in functionality, behaviorally significant computational issues emerge which will be addressed. For this purpose, a temporal transition system for a virtual agent will be presented to introduce fundamental ideas of the algorithms in this section. They are used to provide intuitive access to the parallel execution of parts of the methods. The subsequent sections then extend the model by spatial representations in one and multiple scales.

### 7.1.1 Model and implementation details

In the context of spatial navigation, the MTS  $\mathcal{M}$  as defined in Chapter 5 stores temporal transitions between spatial symbols. Thereby, arbitrary temporal transitions between locations can be learned without access to specific metric information. The only requirement is the detection of a change of location, for instance due to a change of sensor representations.

The MTS  $\mathcal{M}$  consists of two main sub-modules, both of which are subject to a learning procedure. Spatial symbols are stored in the ANN  $\boxed{\Sigma}$ , and temporal transitions in the ANN  $\boxed{\Pi}$ . The typography, i.e. a character within box, is chosen such that the relationship to the corresponding theoretical modules from MTS is immediate, but differentiation between MTS and the implementation itself is feasible.



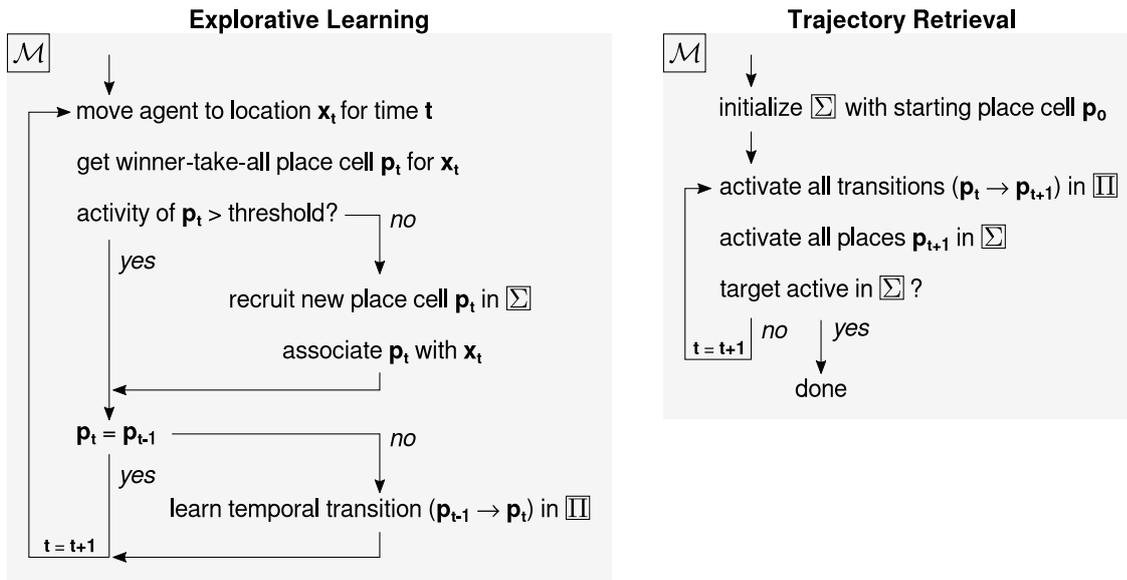
**Figure 7.1 – (Temporal) transition model.** The set of active symbols from alphabet  $\Sigma$  are implemented as a Growing Neural Gas (GNG) which forms an auto-associative memory, whereas the set of transition bundles  $\Pi$  can be considered a hetero-associative network. Both combined form the MTS  $\mathcal{M}$ . The system is indirectly recurrently connected and can receive external input from and send output to other modules, e.g. PFC and motor cortex.

Thus it is possible, for instance, to address the set of symbols  $\Sigma$  and simultaneously discuss the network  $\boxed{\Sigma}$  which acquires and maintains these symbols.

A recruiting process borrowed from GNGs is used to learn symbols [113, 231]. As opposed to most ANNs, a GNG does not require pre-definition of the number of neurons and connections but recruits these as soon as they are required. The generation is usually triggered by some event or error measure which indicates that the already established neurons cannot represent a novel input datum. Thereby, GNGs exhibit principles of self-organization and emergence.

In contrast to regular GNGs which gradually modify weights, a one-shot learning rule with binary weights  $w_i$ , i.e.  $w_i \in \{0, 1\}$ , is used during acquisition of temporal transitions. Furthermore, pruning of neurons, as may happen in regular GNGs, is omitted. The one-shot learning process used in the algorithm is inspired by the results of others and their work concerning associative memories [181, 183, 269]. As long as there is no change in location, the currently active neurons in  $\boxed{\Sigma}$  remain active. Thereby,  $\boxed{\Sigma}$  can be considered to form an auto-associative memory. On the other hand, the functionality of  $\boxed{\Pi}$  follows the concept of hetero-association. An overview of MTS  $\mathcal{M}$  is depicted in Figure 7.1. Potential links to and from other modules such as an artificial PFC or motor cortex are drawn. However, exploration of their impact are left for future work except for the following assumption. PFC supervises the state of active symbols for decision making, e.g. it selects starting locations, monitors if target symbols become active, or to cancel an operation.

Learning of new symbols is triggered by new locations and happens in the following manner. In case that the virtual agent perceives a sufficiently novel location at time step  $t$ , a new neuron  $p_t$  is generated and connected to the current location  $x_t$ . In the model presented in this section, novelty is detected by the distance of all neurons to location  $x_t$ , i.e. novelty is given if  $d(x_i, x_t) \geq d_{\text{thresh}} = 0.1$  m for any neuron  $p_i$  in  $\boxed{\Sigma}$  and associated location  $x_i$ . Thus, each neuron  $p_i$  in  $\boxed{\Sigma}$  corresponds to a spatial symbol and is associated with the symbol coordinate  $x_i \in \mathbb{R}^2$ . The network uses a winner-take all mechanism to select the nearest neighbor for an input location by computing the distance of each  $x_i$  to the current location  $x_t$ . Thereby the neuron



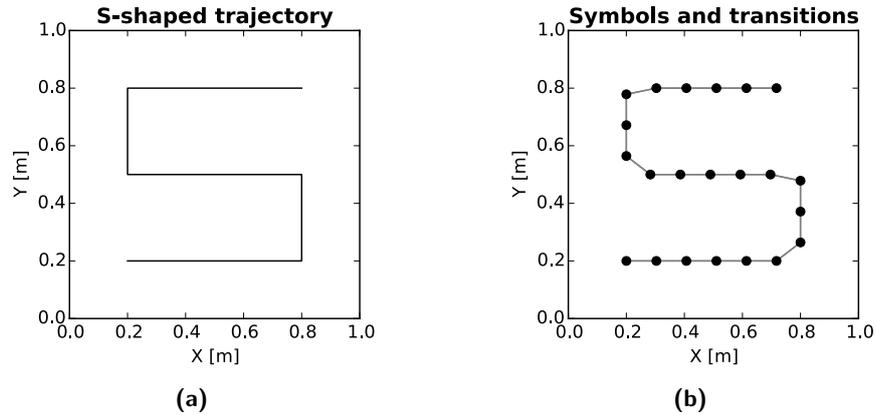
**Figure 7.2 – Algorithms for learning and retrieval in a temporal transition system  $\mathcal{M}$ .** The left hand side of the figure shows a flow chart of the algorithm used during learning. Novel place cells are recruited if the already acquired cells cannot properly represent the input state  $x_t$ . The retrieval of a trajectory is depicted on the right hand side. A transition in  $\Pi$  will only become active if the symbol for which it is defined is active in  $\Sigma$ . Note that the algorithm allows multiple symbols and transitions to become active at the same time.

which represents the closest location to  $x_t$  will express the highest activity. As soon as the currently active neuron in  $\Sigma$  changes, the transition is learned in  $\Pi$ . Thereby,  $\Sigma$  corresponds to the set of symbols  $\Sigma$  and  $\Pi$  to the set of transitions  $\Pi$ . Each neuron in  $\Pi$  preserves directional information, i.e. given a transition  $A \rightarrow B$  it will not learn  $B \rightarrow A$ . Hence, the network learns transitions only through the indirection layer of  $\Pi$ . The overall learning procedure for one time step of the simulation is depicted on the left hand side of Figure 7.2.

During retrieval, any neuron  $p_0$  which corresponds to the starting location becomes active in  $\Sigma$ . Subsequently, the recurrent network is iterated until any neuron becomes active in  $\Sigma$  which corresponds to the target location, or until a maximal number of iterations is reached. Note that the procedure can yield multiple valid sequences of symbols. Furthermore, multiple symbols are allowed to be co-active simultaneously during one iteration, thereby expressing parallelism of computations. Neurons which are active during one iteration cannot become active immediately in the next iteration. A flow chart of the retrieval is depicted on the right hand side of Figure 7.2.

### 7.1.2 Simulation and results

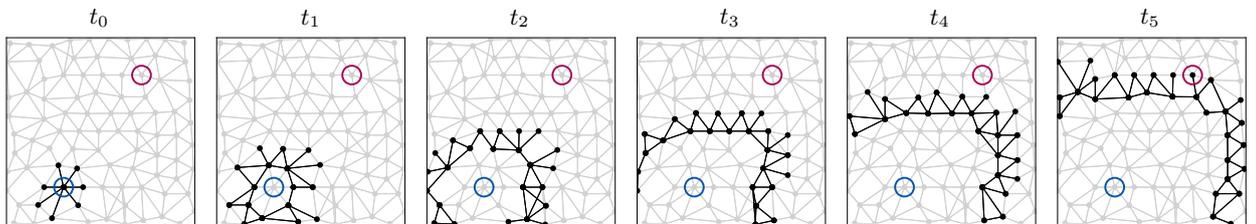
To demonstrate proof-of-principle, the virtual agent was initially moved on an  $S$ -shaped trajectory in a square environment (see Figure 7.3a). Meanwhile, the agent learned several places and temporal transitions from one place to the next (Figure 7.3b) according to the procedure described above. Note that the agent only acquires knowledge about temporal transitions, but not about spatial distances. After activation of the starting neuron,  $\mathcal{M}$  reconstructed the entire trajectory. Figure 7.3



**Figure 7.3 – S-shaped trajectory and replay.** (a) S-shaped training trajectory for the replay experiment. (b) Spatial symbols were created after a minimum distance of  $d_{\text{thresh}}$ . The symbol centers are depicted as black dots, the transitions between symbols as black lines.

shows the trajectory as well as the spatial symbol centers and transitions.

Subsequently, the virtual agent explored the environment with movement statistics similar to real rodents for 20 min (Figure 6.8). Thereby, the agent learned novel locations and transitions. After this additional exploration phase, the network was queried again to recall a trajectory from start to target. As shown in Figure 7.4, all feasible trajectories were explored in parallel. This is visible in form of a wave of active symbols propagating through the network. The number of iterations required to assess the existence of a trajectory dropped from 22 after learning only the S-shaped trajectory to 6.



**Figure 7.4 – Pre-play activity within the network after exploration learning of additional transitions at several time steps.** Given a start symbol (blue circled symbol in lower left), recursive invocation of the retrieval procedure activates all symbols in  $\Sigma$  (black dots) and transitions in  $\Pi$  (edges between dots) until the target symbol (red circled symbol in the top right) is found. Inactive symbols and transitions are marked in gray, active ones in black. Each panel represents one iteration of the entire temporal transition system from start (left) until it found the target (right).

### 7.1.3 Brief discussion of the temporal transition system

The learning procedure generates a connected graph of symbols. However, edges of the graph are given by indirections via transitions. This can be seen immediately in the visualizations given in Figure 7.3 and Figure 7.4. There, each spatial symbol is represented by a black dot and temporal transitions as lines between the dots.

## 7. Algorithmic exploration of the entorhinal-hippocampal loop

The network is able to learn and retrieve trajectories by using only the algorithms presented. What makes the algorithms especially appealing is their simplicity and the inherent parallelism for retrieval of multiple potential goal-directed sequences. This parallelism is demonstrated in Figure 7.4.

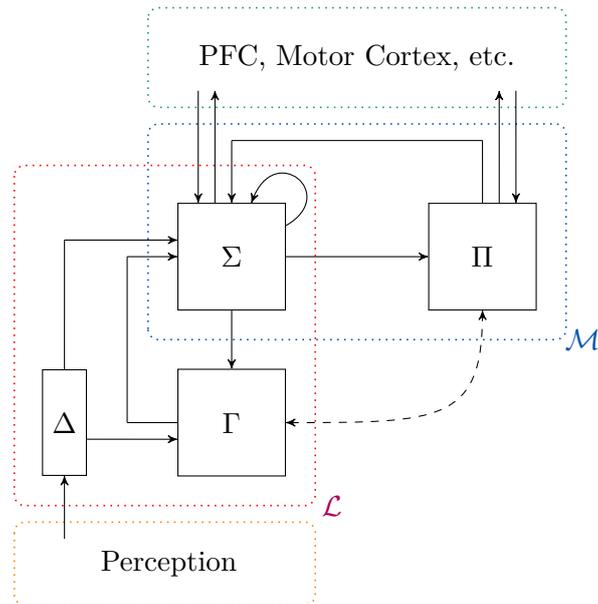
Learning of novel temporal transitions and locations by exploration reduces existence queries of trajectories significantly. However, the model presented here as well as any model based only on learning temporal transitions has a serious drawback. Consider the *S*-shaped trajectory used during the first trail. The shortest distance from start to goal in an open arena would be to cross the arena. As there is no notion of vicinity within the system except through temporal ordering of consecutive locations, the agent has to cover the entire input space to learn feasible shorter routes though. Thus it cannot compute any shortcuts. One possible solution is to incorporate spatial knowledge [99, 100], and perform look-ahead [200]. Both will be addressed in the following sections.

### 7.2 A scale-space model for spatial navigation

A navigating agent equipped with sensors to perceive its environment not only detects temporal transitions from one place to another. In correspondence with its own movement, spatial transitions between locations occur. To include spatial transitions, the algorithms of the previous sections are extended as follows.

The proposed model of entorhinal-hippocampal interactions is depicted in Figure 7.5. In addition to the temporal transition system  $\mathcal{M}$ , spatial transitions are stored in an MTS  $\mathcal{L}$ . Its entirety, i.e. the combination of  $\mathcal{M}$  and  $\mathcal{L}$  is denoted as the spatio-temporal transition system  $\mathcal{P}$ . As discussed in Chapter 6,  $\mathcal{L}$  it is thought to organize due to afferents which can uniquely identify singular locations. There, spatially modulated input was arranged in a regular square grid. The sampling process lead to hexagonal response fields of grid cells, which used spatial sampling in dendritic computations. It was discussed that boundary vector information forms a likely candidate for input representation, as it was used previously for recruitment of place cells [13]. The module responsible for the spatial representation is denoted as  $\Delta$ . Note that the module will operate directly on global coordinates in the algorithms presented below. Future work will address this issue and incorporate an elaborated neural network for spatial identification, e.g. based on the boundary vector cell model by Barry et al. [13]. It is believed that a detailed model is not necessary for the examination of interactions and observation of computational issues presented here, though.  $\mathcal{L}$  is implemented similar to  $\mathcal{M}$ , i.e. a GNG  $\Gamma$  recruits neurons to store spatial transitions between locations. However and in contrast to  $\Pi$ ,  $\Gamma$  not only associates with neurons of  $\Sigma$ . It also requires driving input from its sampling process during learning, represented as the arrow from  $\Delta$  to  $\Gamma$ . The learning and retrieval procedures will be described in detail below in Subsection 7.2.2. In summary, the agent learns spatial transitions from its current position to locations in the neighborhood in addition but independently to temporal transitions. On retrieval, the two systems simultaneously retrieve possible trajectories.

Given sufficient exploration of an environment, the temporal and spatial transition system described here yield approximately the same results for the computation of a trajectory. Each system individually has its benefits from a computational



**Figure 7.5 – Combined spatio-temporal transition model.** The temporal transition model of Figure 7.1 is extended by spatially modulated input  $\Delta$ , which is for instance based on a boundary vector state system. The simulations use a coordinate system for ease of computation, though. Spatial transitions are stored in module  $\Gamma$ .  $\mathcal{L}$  and  $\mathcal{M}$  can operate independently of each other, their combination forms a spatio-temporal MTS. The dashed line indicates a necessary correspondence of spatial and temporal transitions.

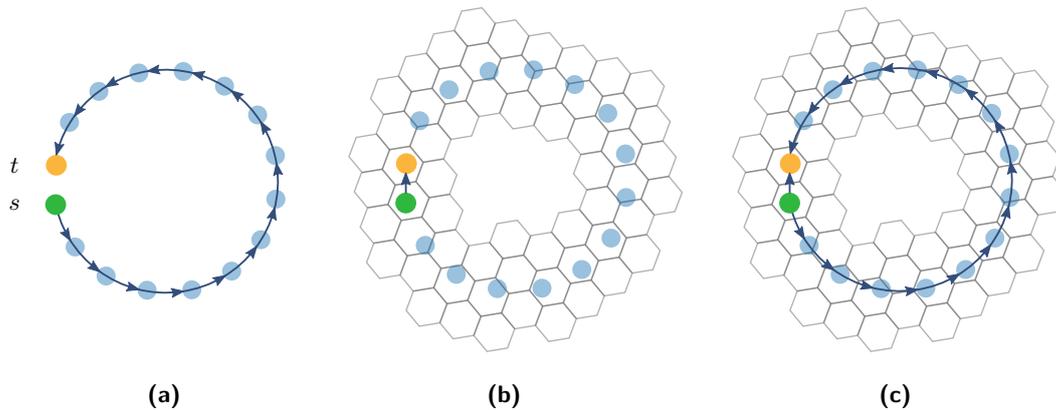
perspective, though.

The spatial transition system provides a benefit when solving issues related to loop-closure. These problems are concerned with the detection when a location is re-observed. Consider an agent that walks on a circular trajectory in which the start and goal locations are spatial neighbors as depicted in Figure 7.6a. Using only a temporal transition system, the agent always returns the entire trajectory if asked for the shortest path from start to goal. As it is unaware of the spatial relationship between the start and the target location, it cannot find a shortcut. Given a system to learn spatial transitions to the neighborhood of perceived locations, the agent is able to determine that the goal location is in the spatial neighborhood of the start location.

Learning a spatial neighborhood in a biologically plausible network could be achieved by recurrent drive. This assumption is based on observations made in the model to examine the STDP decorrelation hypothesis in Chapter 6. There, recurrent activation from transition neurons was able to drive neurons which were sub-thresholdly stimulated before to their spiking threshold. Thereby, transition neurons learned and increased weights to neurons which represented related information. Thus, it is assumed that place cells which are inactive but closeby are likely to get activated by a spatial transition system. Note that there may be a restriction of learning spatial neighborhoods due to the necessary temporal co-activation of place and grid cells. This issue will be addressed further below.

Using only a spatial transition system has a significant drawback. As visualized in Figure 7.6b, the spatial transition system may not be able to retrieve a temporally

## 7. Algorithmic exploration of the entorhinal-hippocampal loop



**Figure 7.6 – Example for temporal, spatial, and spatio-temporal transition computation.** (a) Given a circular trajectory (black line) from start  $s$  (green circle) to goal  $t$  (yellow circle), a temporal transition system only learn immediate transitions from one location (blue circles) to the next. On recall, the system reproduces the whole sequence. (b) A spatial transition system learns transitions in form of spatial neighborhoods. The system does not learn a sequence of visited places. A spatial symbol is indicated as hexagon, its neighborhood are the surrounding six hexagons of each spatial symbol. The blue circles are the places of (a) overlaid on the spatial information. In the algorithm, places are associated with their closest spatial symbols. On recall, the spatial transition system is potentially able to find a short-cut. Depending on the trajectory, the system is not guaranteed to reproduce a temporally coherent sequence, though. (c) The combined spatio-temporal transition system is able to reproduce the real sequence of places. Furthermore it can find shortcuts in a spatial neighborhood.

coherent sequence of locations. In the depicted example, the spatial transition system learned neighborhoods along the original path (superimposed on the image by blue circles). However, when queried for a path from start to target, it will report the shortest path. Even if there was no immediate short-cut from start to target, any transition to neighbors is equally likely. Thus, the spatial system will report multiple viable trajectories but not reproduce the original temporal sequence. Therefore a temporal transition system is necessary for the purpose of learning and retrieving an actually performed sequence of steps.

There are two main reasons to suggest a separation of spatial and temporal transition systems. In case that multiple viable trajectories are returned by the system, a winning trajectory should be selected by the agent. For this purpose, the agent has to record the sequence of locations and actions that it actually performed as well as any reward that it received while doing so. Thereby, a binding of motor action and rewards has to occur. The binding can be implemented in a biologically plausible manner as demonstrated by Hirel et al. [146]. The authors suggest that the binding occurs in temporal transition neurons, which store motor commands and rewards. They argue that the reward signal arises due to dopaminergic neurons between striatum and substantia nigra pars compacta (SNc). The latter is part of the substantia nigra, which is located in the midbrain. Furthermore they argue that connectivity across the lateral hypothalamus could transport reward signals from SNc to the Hippocampus.

The second reason for the separation is due to considerations of fault tolerance and

abstraction, and uses a perspective from computer science. If the spatial and temporal transition systems were merged into one singular system, any change to the sensory representation would likely affect the internal representations of transitions and, thereby, the reconstruction of temporal sequences. On the other hand, a separation of spatial and temporal transitions allows grid cells to act as mediator between sensory states, place cells, and keeps spatial information separate from actually performed sequences. This is similar to abstraction layers and design patterns known and widely-used in Object-Oriented Programming (OOP) [119]. Only the sampling process, which was suggested to be a dendritic computation performed by grid cells in Chapter 6, has to adapt in the case that the sensory representation changes. The spatial neighborhood information encoded in the connectivity from grid to place cells remains stable. Furthermore, a single sensory representation can be used via grid cells in multiple configurations of place cells. Thus, a modularization of temporal and spatial transitions is suggested to increase representational capabilities and fault tolerance.

### 7.2.1 Multiple scales and the algebraic number $\sqrt{2}$

Recall one of the primary computational tasks of the spatio-temporal MTS  $\mathcal{P}$ . The goal is to compute the existence of one or more viable trajectories from a start to a target location and select the most appropriate one. Alas, the spatio-temporal MTS presented above has two behaviorally significant issues.

Consider an animal that wishes to compute a linear path from a start  $s$  to a target  $t$ . Let the activation of symbol and transition neurons require 5 ms each due to axonal delays. This means that neurons corresponding to subsequently possible locations will activate only after 10 ms. Furthermore, let the distance between start and target  $d(s, t) = 200$  m and let a transition between spatial symbols cover a distance of approximately 10 cm. It is required to iterate the entire transition system about 2000 times until the target is found. Hence, the total time to query just the existence of a trajectory already takes approximately 20 s. Certainly an amount of time in which the animal falls an easy victim to predators. The problem is further amplified when trajectory selection is included, for instance by reward propagation. An acceleration technique is clearly required.

The second issue is closely related to the first. In the example with the circular trajectory of the previous section, local neighborhood information provided by grid cells improves the time to travel from start to goal significantly and can lead to shortcut detection. However, a problem arises as soon as the start and the end goal are not immediate neighbors. Larger distances cannot be interpolated, and thus discovery of novel shortcuts is limited.

One reasonable candidate of acceleration is to perform some form of look-ahead. Instead of only computing transitions of immediate temporal neighbors, the system could also learn transitions across several temporal distances. Furthermore, spatial look-ahead could be performed in unison to temporal look-ahead. Recall that grid fields are proposed to be result of dendritic computations of grid cells, i.e. to detect a transition from one location to another, they have to be able to detect locations. Consequently, the response fields should increase when spatial look-ahead is performed. It is thus necessary to first define the sampling mathematically and discuss the meaning of the size of a sampling process to describe look-ahead properly

and determine the ideal increment of grid field sizes.

On the smallest scale, the transition system has to distinguish two consecutive spatial locations both for temporal as well as spatial transition coding. Certainly the resolution of a spatial sampling process cannot be made arbitrarily small in a finite system and is likely associated with uncertainty. The spatial sampling process is therefore modelled as a normally distributed Probability Density Function (pdf), resembling the finding that tuning curves of sensory neurons often express a bell-shaped form [43, 168]. Furthermore, it is assumed that a dense sampling arrangement is available as discussed in Chapter 5 and Chapter 6. The pdf is defined as a Gaussian function, formally

$$G(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right). \quad (7.1)$$

The function is parametrized by mean  $\boldsymbol{\mu}$  and an  $n$ -dimensional co-variance matrix  $\boldsymbol{\Sigma}$ .  $T$  denotes the transpose of a vector or matrix, and  $^{-1}$  indicates matrix inversion. Choosing a Gaussian function has several benefits which will be discussed later.

The resolution of the sampling process is expressed by  $\boldsymbol{\Sigma}$  and depends on the number of samplers available and the size of the space that has to be sampled. The concept of a smallest feasible resolution can be denoted as follows.

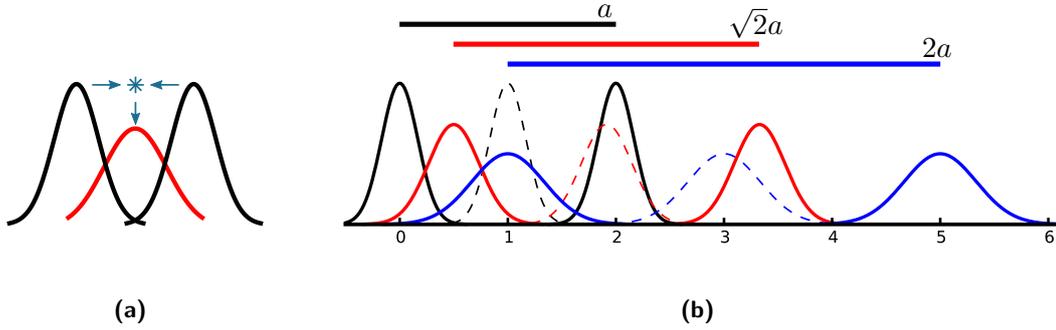
**Definition 5** (Eigenresolution). *The eigenresolution is the minimal resolution of a transition system and is characterized by the covariance matrix  $\boldsymbol{\Sigma}_{\text{eigen}}$ . In the case of a rotational symmetric receptive field,  $\boldsymbol{\Sigma}_{\text{eigen}} = \sigma_{\text{eigen}} \mathbf{I}$ .*

Different transition systems may expose independent eigenresolutions. It appears unlikely that an animal requires accurate measurements in the millimeter range for path planning during spatial navigation. Rather, the resolution is assumed to be limited within the approximate size of the animal itself. In other systems where higher or smaller accuracy is necessary, this number may change accordingly.

Temporal transitions can be understood to form an ordered list of data. Due to the temporal distance based on spatial sampling parametrized by  $\boldsymbol{\Sigma}_{\text{eigen}}$ , any two consecutive items in the list are approximately equidistant in time. From computer science it is known that the asymptotically optimal search strategy for such data is binary search, which will improve time to search exponentially [185, 6.2.1]. Hence temporal distance should increase by a factor of 2 to gain exponential speedup and thereby provide optimal temporal look-ahead.

In the worst case of a linear path, temporal and spatial transitions coincide. Thus, the optimal search strategy for spatial look-ahead follows the result for temporal look-ahead. Thereby, the spatial sampling process for spatial look-ahead has to adapt accordingly to recognize the combination of two locations. It is noted that the following derivation considers *exclusively* the on-center portion of one receptive field of the suggested grid cells. Note however that it is assumed that the entirety of receptive fields can be modelled as a stitching together of Laplacian of Gaussians (LoGs) (also known as *Mexican Hat* functions), but this will be left for a future study.

To construct spatial look-ahead, two consecutive locations have to be merged. They are represented by the two corresponding sampling processes, modelled as normally distributed pdfs and parametrized by  $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_1$ , and  $\boldsymbol{\Sigma}_2$ . Note that combining



**Figure 7.7 – Scale space construction and multi-scale response fields.** (a) The convolution of the response of two spatial samplers (black curves) forms the spatial sampler of the next scale (red curve). The convolution is indicated by the arrows and the asterisk symbol. (b) Transition cells respond at the same time as their spatial sampling process (black curve, first scale). However, they must decorrelate from immediate perceived neighbors on the scale (black dashed curve) to fulfill the constraints given by MTT. Every subsequent scale has receptive fields which are  $\sqrt{2}$  times larger than the previous scale (red and blue curves). Also, their spatial period increases by a factor of  $\sqrt{2}$ . The distance between periods are depicted by the bars above the curves.

distant locations would violate the coherency constraint of MTT. Merging of spatial samplers corresponds to the convolution of the two pdfs. The novel coordinate system is centered between the previous two locations *by construction* (see Figure 7.7a). It can be shown that convolution of two  $n$ -dimensional normally distributed pdfs results in another  $n$ -dimensional normally distributed pdf with combined mean and co-variances, i.e.  $\boldsymbol{\mu} = \boldsymbol{\mu}_1 + \boldsymbol{\mu}_2$  and  $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2$  (see for instance the supplementary material to Vinga et al. [368] which presents the entire derivation of the multi-dimensional mean and variance). Given the assumption of uniform samplers and symmetry in two dimensions on the smallest scale, i.e.  $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_{\text{eigen}}$  and  $\boldsymbol{\Sigma}_{\text{eigen}} = \sigma_{\text{eigen}}^2 \mathbf{I}$ , it follows that the variance  $\sigma^2$  of a sampling process for spatial look-ahead is  $\sigma^2 = 2\sigma_{\text{eigen}}^2$ . Thereby the integration area of a spatial sampling process which integrates two locations is twice as large as the integration area of the previous scale. It also means that the radius of the on-center field increases by  $\sqrt{2}$ . The off-center area, which corresponds to the feasible target region of a spatial transition, increases accordingly. The densest packing of the center-surround fields of a transition neuron remains optimally at a hexagonal arrangement, however their spatial period is increased by  $\sqrt{2}$  according to the increase of the spatial sampling process (see Figure 7.7b).

So far, one additional scale was formed by combining two spatial transitions from the previous (smallest) scale. In accordance with binary search for temporal look-ahead, the method is applied *recursively* to form consecutive spatial scales. The result is a discrete increment in grid field sizes which show twice the integration area, and an increase along each dimension and thus their period by a factor of  $\sqrt{2}$ . The increment is visualized in Figure 7.7b. Data suggests that grid fields indeed increase along the dorso-ventral axis [336]. Interestingly, the observed field sizes were discretized by a factor which is close to  $\sqrt{2}$  when comparing two consecutive scales.

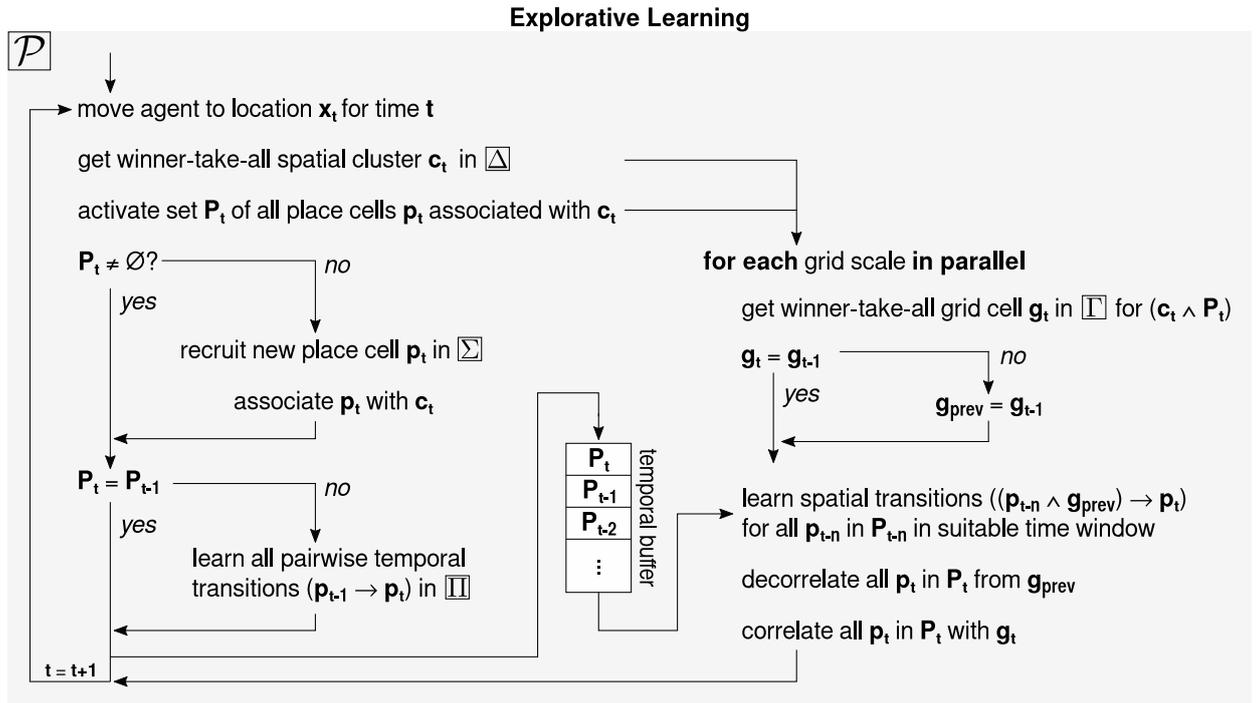
### 7.2.2 Simplified model and results

The theoretical results obtained above were simplified further to demonstrate proof-of-principle. The arrangement of spatial samplers was pre-defined to be arranged hexagonally and form Voronoi cells. Thereby, the cluster region of a sampler is also hexagonal. Furthermore, the spatial sampler which is closest to the current location is thought to represent the input space the best according to a winner-take-all mechanism. Hence, the space in which the agent navigates is already discretized according to the hexagonal arrangement.

Due to the absence of additional afferents, the spatial sampling process was used to detect spatial as well as temporal transitions and also to recruit place cells. In contrast to the temporal transition system  $\mathcal{M}$  of Section 7.1, the recruiting process of place cells depended on the activation of a spatial sampler instead of directly measuring distance within the abstract place cells. In other words, as soon as a spatial sampler was activated in  $\boxed{\Delta}$ , the best matching unit in  $\boxed{\Sigma}$  was determined. If the activity was below a certain threshold, a novel neuron was recruited. Although the activation of the samplers themselves were computed as Euclidean distance and the threshold set to a distance of 0.05 m for the purpose of simplification of the model, the place cells were thereby independent of the underlying metric. Hence the model could be altered by exchanging the computation of activity as a result of Euclidean distance to neural activation, for instance given by boundary vector input.

The spatial sampling centers were used to form grid cells in multiple scales. The scale increment was set to  $\sqrt{2}$  according to the theoretical results described above. Note that the discretization and pre-computation of grid fields has several numerical issues. To examine their impact, the grid fields were arranged such that parts of the *S*-Shaped trajectory, already used in temporal transition system  $\mathcal{M}$  (see Figure 7.3), fell on the apex of intermediate sampling clusters. This meant that the agent's perceived location, which was represented by the discrete sampling process, was prone to jump vertically up- and down according to the sampling centers while the agent actually moved only horizontally. Note that these effects are the results of the simplifications introduced here, primarily the discretization. It is expected that an elaborate model involving non-discretized sampling fields is unlikely to produce similar artifacts and will be subject of future studies. It is believed that the discretized variant suffices to demonstrate proof-of-principle, though.

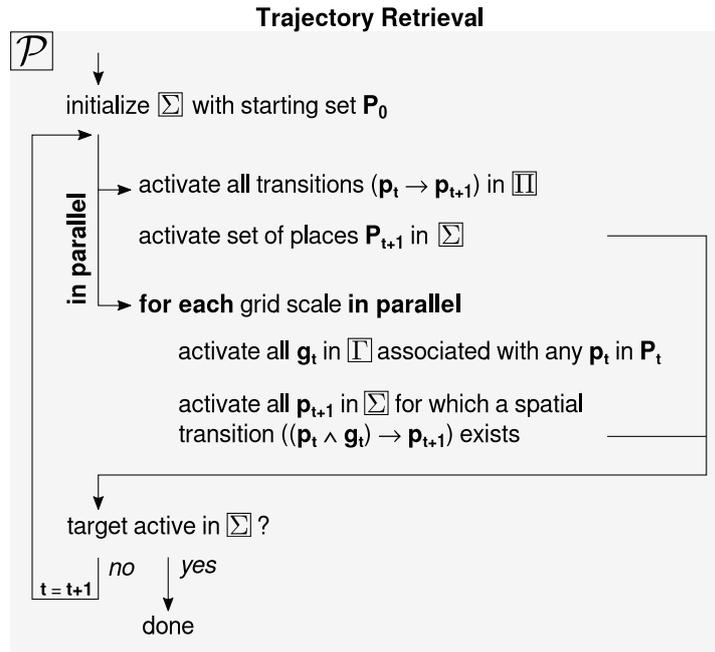
Figure 7.7 illustrates the learning algorithm of the scale-space model during exploration. Note that acquisition of place cells and temporal transitions is similar to the algorithm used in  $\mathcal{M}$ , however presented here in a way which allows multiple simultaneously active place cells  $p_t \in P_t$ . It is further extended such that place cell activity is buffered in a temporal buffer structure. Buffering is required during acquisition of multiple scales of spatial transitions because the transitions are only learned if their corresponding spatial afferents are in a suitable integration window. Consider a single grid cells. It is selected as best matching unit during the winner-take-all selection process according to the sampling center  $c_t$  with the highest activity in combination with pre-synaptically active place cells (if any). Inspired by descriptions of synaptic circuits [323], the co-activation used here is formed by a simple *logical and* operation. Recall that place cells are recruited by the spatial sampling process on the smallest scale, though. Thus, a grid cell on larger scales expresses on-center regions



**Figure 7.7** – Algorithm for learning transitions in a spatio-temporal transition system  $\mathcal{P}$ .

which may cover multiple place fields. Learning is therefore modelled according to the following process. Any place cell  $p_t \in P_t$  which is co-active with  $g_t$  is correlated with  $g_t$ . In contrast, any place cell  $p_{t-n}$  which was in a suitable previous temporal window and co-active with a previous grid cell  $g_{\text{prev}}$  will be decorrelated from  $g_t$ . Note that in the presented simplified model, the correlation and decorrelation mechanism reduces to a binary operation of tagging a place cell to be co-active with either one grid cell or another. Spatial transitions are stored as a *logical and* connectivity between any previously active place cell  $p_{t-n}$  and the previously active grid cell  $g_{\text{prev}}$  alongside any currently active place cell  $p_t$ . The suitable temporal window to select  $p_{t-n}$  for the acquisition of spatial transitions depends on the scale. On the smallest scale, only immediate temporal neighbors are considered. Due to the spatial scale increment of  $\sqrt{2}$ , the number of place cells in the suitable time window increases by 1 for every next scale. The learning procedure is drastically simplified in order to omit explicit modelling of potentially non-linear temporal dynamics of grid cells and their temporal integration windows. Nevertheless it is believed that it captures the algorithmic effects of temporal integration sufficiently well. A model involving STDP and spiking neurons to examine biologically accurate transition learning is currently in development but will be left for future work.

Retrieval of trajectories is performed as visualized in Figure 7.8. Note the difference to learning. Spatial transitions are only learned when both the location is within a spatial cluster as well as when it happened in the temporal integration window of the corresponding grid cell. Thereby, spatially adjacent locations will not automatically learn a spatial transition if there is no temporal correspondence. This is another severe simplification of a learning rule which depends on spike timings, but suffices

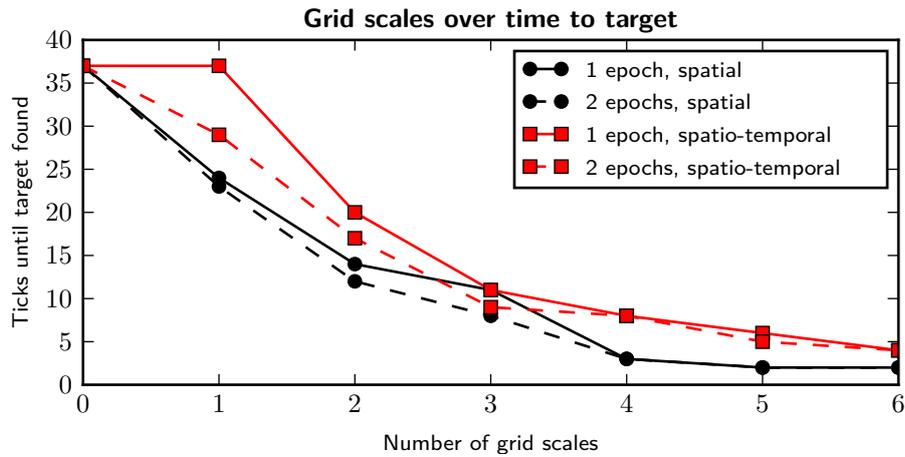


**Figure 7.8 – Algorithm for retrieval of transitions in a spatio-temporal transition system  $\mathcal{P}$ .**

for demonstration purposes. Furthermore, learning will associate place cell activity with a sensory representation according to co-activation learning, which is expressible as a *logical and* operation. During retrieval, a spatial transition from one place cell to another requires the co-activation of the previous place as well as the activation of the grid cell which is associated with the previous place. However, activation of grid cells happens for any previously active place cell, which corresponds to a *logical or* operation. In other words, place cell activity can drive grid cells without sensory inputs during retrieval. If not stated otherwise, learning of transitions is continuous during retrieval. The reason is that retrieval may generate novel sequences of spatially adjacent locations which were previously restricted by temporal progression.

The following protocol was used to observe the impact of the scale-space representation and of continued learning during replay. The *S*-shaped trajectory was presented to  $\mathcal{P}$  once to initially learn places and transitions. Afterwards, the system was queried to reconstruct the entire trajectory while learning was turned off. Then, the trajectory was replayed. During replay, future places were allowed to be pre-fetched and stored in the temporal buffer. In contrast to exploration, replay was based only on spatial and temporal transitions, i.e. successive activation of spatial symbols without access to true sensory states. Subsequently, the system was queried again to record if replay lead to the detection of novel transitions, and thereby potential short-cuts. In addition to these two settings, the impact of the temporal buffering was examined. The learning and retrieval strategies were repeated, but the temporal integration window was ignored during the learning.

The results of the protocol, which results in a total of four configurations, are depicted in Figure 7.9. The network learned 38 places on the *S* shaped trajectory from start to goal, which resulted in 37 iterations of the entire system until the



**Figure 7.9 – Computational times of the scale-space model.** The model was applied with varying number of scales to an  $S$  shaped trajectory, for which 38 individual place cells were created. The black bullets are the number of iterations (ticks) until the target location was found using only spatial pooling. The red squares are the ticks required when spatio-temporal correlation was taken into account. Solid lines indicate data after one-shot learning. Dashed lines indicate the results when one additional training was used during which place cell replay was induced by grid cell activity.

target was found when only temporal transitions were acquired. The first grid scale was without impact on the computational performance if the temporal window was enabled. When temporal buffering is ignored, already the first grid scale improves computational performance. Further scales improve computational times according to an exponential decay in the number of iterations required until the target was found.

### 7.3 Discussion, observations, predictions

In the following, general remarks on the model and its results will be stated. Subsequently, some of the results are examined in more detail with respect to biological findings and relationship to other fields of research. Finally, an outlook to currently ongoing and future work is given.

The integration area for spatial look-ahead on large scales does not necessarily correspond to the temporal look-ahead due to the optimal scaling factor of  $\sqrt{2}$ . This can be observed in Figure Figure 7.7b. There, the blue line corresponds to the second spatial look-ahead scale. Intuitively, it could be assumed that it integrates four locations. However, the integration area covers only three consecutive locations due to the optimal construction of the scale. This observation leads to consequences for learning multiple spatial scales. All locations in a sequence have to be buffered temporally to allow consecutive access to form the spatial scales in case that learning is based on associative learning without random access to previously learned locations, e.g. via an STDP learning rule and co-activation of neurons. Conversely, the number of scales which can be learned in such a manner depends on the number of elements in the temporal buffer.

## 7. Algorithmic exploration of the entorhinal-hippocampal loop

The emerging pyramid of the spatial sampling process described in Subsection 7.2.1 can be understood in the following way. A singular location is represented on the finest resolution by spatially modulated neural activity, for instance boundary information. Sampling from this representation on the smallest scale corresponds to an *identity* operation. The state of activity can be understood as a spatial descriptor, as it represents a single location. The target region of the transition is also characterized by spatially modulated neural states, thereby forming a spatial descriptor for an entire region. To construct additional scales for spatial-look ahead, the descriptors are low-pass filtered according to a convolution operator by which detailed structures represented in the neural state are removed. The smoothed descriptors on larger scales allow to perform location comparison and thus transition prediction and look-ahead on distances which increase according to optimal search. Consequently, computational run-time for trajectory planning is sped up exponentially.

The described method of Subsection 7.2.1 and interpretation presented here is the construction of a Gaussian pyramid (or Laplacian pyramid when receptive fields are assumed to be LoGs), well-known in the signal and image processing communities [213, 215, 385]. There, scale-space theory was developed to improve image understanding [216, 219], compression [8, 41], and modelling of retinal or visuocortical receptive fields [19, 120, 392]. The approaches operate according to the following principle. Coarser scales represent an original signal by reduction of complexity and simplifications of structures embedded in the signal [187, 213, 385]. In image processing, this corresponds to smoothing – or generally speaking filtering – an image, thereby low-pass filtering it and removing fine-scale information. According to Lindeberg [214], Gaussians and their derivatives are suitable convolutional operators to form a scale-space representation. They invariant to several transformations, e.g. rotation, do not introduce local minima or extrema during the convolution. The latter could lead to erroneous behaviors. Reduction of details of the input signal is essential to allow feature detection across multiple scales [220]. Certainly, this sounds appealing for spatial navigation. To improve computational performance or find shortcuts, spatial and temporal neighborhoods have to be determined over larger distances.

In computer science, subdivision of dimensions and operation on simplified representations is a common practice to speed up computations and were often proven to be optimal for their respective task. For instance, kd- [20], quad- [108, 158], and octrees [267, 310] are tree data structures to exponentially speed up performance when searching spatially arranged objects. The suggested mechanism expressed by grid cells, though similar to such tree data structures, has the benefit that it does not require a global coordinate system. The low-pass filtering and sub-sampling of spatial responses as well as the detection of transitions on approximately scaled representations of locations is proposed to yield *local estimations* of spatial neighborhoods. This can also be understood in the following compatible way.

Consider a graph theoretical notation for symbols and transitions. In such a notation, symbols correspond to vertices in a graph, and transitions are represented by edges between the symbols. Temporal transitions give rise to edges which form a temporal sequence. Spatial transitions allow to also store local out-bound transitions from one symbol to another. Furthermore, the introduction of multiple scales allows to additionally connect each symbol to other symbols which are further away than

just the local neighborhood. In other words, each symbol has the capability to *participate* in transitions on larger scales. Only local knowledge and similarity of the symbols are required to introduce the links, no global information about the exact location of each symbols is necessary. Consequently it is proposed that the hippocampal-entorhinal loop forms a scale-space MTS, expressed by the activity of place and grid cells. The cells thereby form a topological space, suggested already in [75, 76, 149].

The results show that a scale-space representation of spatial transitions leads to an exponential improvement of computational times. Note that the *S*-shaped trajectory is almost linear on smaller scales, i.e. almost no perceivable bends or curves appear, which is considered to be the worst case scenario for the described technique. As soon as non-linear segments appear, or when the scale is increased sufficiently, shortcuts *around corners* can be detected which reduce the computational times, though. This can be observed, for instance, in the difference between learning only one epoche and additional learning due to replay of the sequence. In the latter, consecutive places which are spatially but not necessarily temporally close get activated. The result are shortcut transitions which are detected only due to co-activation of spatially close places and without sensory information, and thereby improved computational times. Note that grid cell based spatial look-ahead was suggested previously by Kubie et al. [200]. The authors used multiple scales of grid cells to steer a simulated agent towards a goal. The largest scale was used to approximately locate the target location and drive the robot towards the goal. The target location was then successively narrowed down using smaller scales. However, the solution proposed by the authors does not guarantee the consistency and coherency constraints of the final trajectory. Furthermore, the authors did not explain the origin of the discrete spatial scale progression. A similar approach has been suggested by Edvardsen [93]. He proposed a passive mechanism which shifted activity within a network of grid cells of multiple scales towards a target state. Likewise the model by Kubie et al. [200], coherency of the sequence is not guaranteed, nor is the scale increment discussed.

It is suggested that additional speed-up can be gained when the impact of temporal buffering and the temporal integration window is reduced, for instance by a time-compressed representation of locations. Recall that real biological networks are commonly assumed to be subject to STDP learning rules [23, 78, 393]. In these rules, synaptic weights increase when pre-synaptic spikes are in a suitable time-window before the post-synaptic neuron spikes. Usually, the time-window is small and in the order of 10 – 100 ms [121, 186]. Consequently, neural states for places which are far apart have to appear within the temporal integration window of a grid cell to learn the transition between these places. Thus, a temporally compact representation of locations is required to allow neurons of very large grid scales to associate with their corresponding inputs. The necessary temporal compression appears to happen in real rats during SWP-R [44, 45, 72, 164]. During SWP-R, temporal sequences of perceived places are replayed both in forward as well as backward order of their appearance during exploration. Hence it is likely that large scales form on the basis of SWP-R. The model thus predicts a subtle difference of the neuro-dynamics of grid cells on different scales. Learning of spatial transition on larger scales is restricted to a suitable temporal integration window, which needs to be reflected both in terms of a temporal buffer as well as a slight change of the temporal association mechanism.

## 7. Algorithmic exploration of the entorhinal-hippocampal loop

Specifically, the integration time windows of cells on larger scales have to be slightly increased to allow co-activity learning of multiple places according to the scale.

The effect of multiple scales to computational times is limited by the length and shape of the trajectory. In the data shown in Figure 7.9, there is no significant improvement after addition of a sixth scale. Improvement due to multiple scales re-appears only as soon as very long trajectories are presented to the system. Hence it is considered likely that there is a match between the number of scales, the explorative behavior of an animal, and the size of its habitat. However, association of places to large grid scales is subject to temporal buffering and compression as mentioned above. Disruption of SWP-R is therefore suggested to reduce the representation of spatial transitions. Conversely, replay of place sequences is assumed to consolidate spatial knowledge. This prediction was confirmed during writing of the manuscript by Roux et al. [302] who showed that SWP-R indeed stabilizes the spatial map of real rodents.

Temporal and spatial transitions coincide almost exactly on the smallest scale of the model. This is only true if place cells are considered to form only on the basis of spatial input, though. Evidence suggests that this is not the case and that other cues, for instance odors, have a role in the recruitment of novel place cells [97, 395]. Furthermore, multiple place cells may be recruited at the same behaviorally relevant places to over-represent the location.

The scale-space MTS presented here has a clear distinction between learning and retrieval. During learning, a logical AND operation was used, whereas in retrieval a *logical or* computation is performed. It is expected that the operations are implemented as a hetero-synaptic circuit which can be toggled in some way. Therefore it is likely that learning or retrieval can be suppressed independently of each other in studies.

The algorithms presented in Figure 7.7 and Figure 7.8 cannot learn or determine arbitrary shortcuts. Locations which are not within the temporal buffer or in a compressed representation cannot be joined using only the described techniques. This is in agreement with findings during psychological studies, in which rats were not able to compute shortcuts in such scenarios [132]. Furthermore, there is no difference between re-play and pre-play of a trajectory in the scale-space MTS. It is therefore suggested that the pre-play observed by Pfeiffer et al. [282] is due to the same mechanism as re-play during SWP-R. The authors recorded place cell activity in awake animals after they explored an environment for food and had to plan a trajectory to their home location. The authors found that place cells which corresponded to the subsequently chosen trajectory were more likely to get activated in order than not.

The scale-space model and its spatial sampling process predict place cell activity which is not randomly distributed in the absence of non-visual afferents. More specifically, it is suggested that removal of non-visual cues will lead to peak activity of place cells which correlates strongly with a hexagonal arrangement. Preliminary manual clustering of place cell data of real rats suggests that this is indeed the case. However, further data needs to be evaluated for a conclusive statement.

On basis of the obtained results, the following functional levels of the entorhinal-hippocampal loop are proposed and depicted in Figure 7.10. Perceptual systems provide a sensor space that is suitable for localization, the latter which is computed in

place cells. It is proposed that discrete scales of grid cell responses, which were already observed in recordings by Stensola et al. [336], generate a scale-space representation of spatially modulated descriptors and encode spatial transitions. Similar to vision, where scale-spaces are used to detect features across scales [214,216], their purpose is proposed to define spatial transitions to neighborhoods on larger distances and to find shortcuts across larger gaps. Thus, they are a means for spatial look-ahead and, simultaneously, perform *spatial pooling* of related inputs across scales, and reduce temporal execution of trajectory computation exponentially. Certainly the association of place cells with co-active spatial transition cells on multiple scales requires access to potential future and factual past places. Generally speaking, it was noted by Lindeberg that any scale-space system requires access to a temporally buffered data [214]. This in turn suggests a link to Theta phase precession, which is understood as a temporal buffer mechanism by some authors [188,246]. In addition, temporal transitions are stored to record performed actions and potentially associate transitions with rewards, as suggested by Hirel et al. [146]. The latter will be examined in a future study. It is suggested that the association to multiple inputs performed by grid cells is based on a co-activation learning rule. Such a rule was recently used to induce realignment of grid cells given sensory cues [258], i.e. the re-orientation of the response of the cells and their preferred orientation with respect to a global coordinate system.

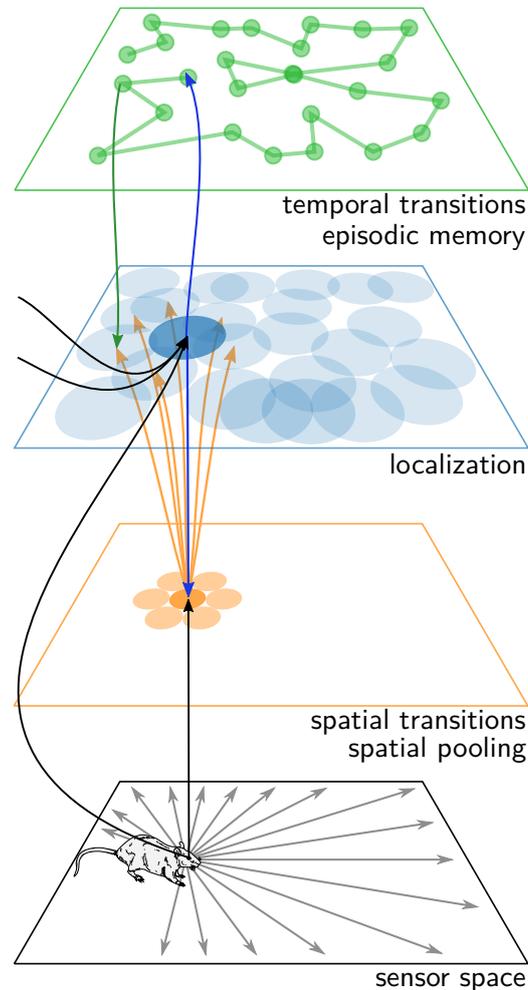
### 7.3.1 Temporal buffering, Theta phase precession, and number of scales

The proposed scale-space model requires temporal ordering and buffering of events. This *post-dicts* observations that excitatory drive from the Hippocampus is required for grid cell activity [28]. The reason in the proposed scale-space model is the necessary synchronization mechanism for grid and place cells to solve the problem of binding places to larger grid scales. In other words, grid cells of larger scales are predicted to require access to spatial symbols (place cells) in their pre-spike time window when learning is performed via STDP.

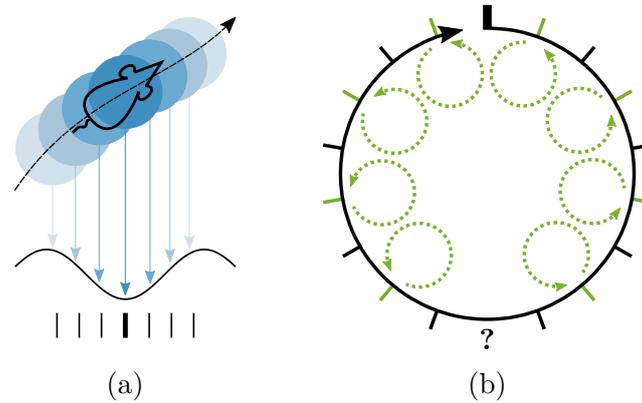
An observable effect which is considered to be the result of a temporal buffering mechanism is Theta phase precession [169], depicted in Figure 7.11a. During motion, not only the place cells which represent the current location of the animal demonstrate spiking activity, illustrated by the dark blue circle in the figure. In addition, several place cells corresponding to places before and after the current location show increased levels of activity, depicted as light blue circles. However, their time of spiking is relative to Theta, a regular oscillation observable in the hippocampal formation [48]. The temporal order of places during Theta phase precession is preserved. The reported compression ratio is up to an order of 10 : 1 [328], i.e. within one Theta cycle up to ten cells spike with a relative temporal shift to Theta. It was already reported that temporal compression is likely to improve Hebbian learning and thereby plays a critical role in temporal sequence learning and memory consolidation [242].

It is proposed that repeated iteration of the temporal transition system  $\mathcal{M}$  can be related to Theta phase precession. Iterating  $\mathcal{M}$  will simply yield the temporal progression of observed locations. Assume a short temporal latency of about 5 to 7 ms until neurons representing spatial symbols and temporal transitions reach their spiking threshold. One iteration of the  $\Sigma$ - $\Pi$ -loop of  $\mathcal{M}$  will thus take about 10 to 14 ms. Given that Theta oscillates at up to 10 Hz [221], it is possible to fit up to 10

## 7. Algorithmic exploration of the entorhinal-hippocampal loop



**Figure 7.10 – Functional levels of goal-directed navigation and localization.** Perceptual systems provide a *sensor space* (gray arrows) which is able to uniquely identify spatial locations (black arrow from *sensor space* to *localization*). Furthermore, a self-organization process forms grid cells on the basis of the sensory space (black arrow from *sensor space* to *spatial transitions*), which can be used to learn and encode spatial transitions from one place to its spatial neighbors (recurrent interaction of *localization* and *spatial transitions* across the blue and orange arrows). Thereby, grid cells perform *spatial pooling* of all place cells which are active for a certain configuration of the *sensor space*. Furthermore, temporal transitions are stored within an episodic memory which allow to retrieve the actually performed transition (recurrent interaction between *localization* and *temporal transitions* along blue and green arrows). Place cells (*localization*) also form on additional, non-visual cues (additional black arrows leading to *localization*), however they are currently not considered in the algorithms.



**Figure 7.11 – Loop nesting and Theta phase precession.** **(a)** Illustration of phase precession during movement. Place cells corresponding to locations before and after (thin black bars) the current location (thick bar) spike relative to the Theta oscillation (black sine wave). **(b)** Change of representation of the sine wave yields a *main loop*. Several  $\Sigma$ - $\Pi$  loops (green circles) of  $\mathcal{M}$  can be nested within one Theta cycle (black circle). The black bars correspond to activity of neurons in  $\Sigma$  and the green bars to activity in  $\Pi$ . The location marked with ? indicates the point of reversal of directions, i.e. neural activity (thin black bars) before this point corresponds to outbound locations of the current place (thick black bar) and therefore pre-play of immediate locations, whereas activity (thin black bars) after the ? are in-bound locations and therefore re-play.

nested  $\Sigma$ - $\Pi$ -loops into one Theta cycle. By this construction, Theta corresponds to a *main loop* which resets the activity of the temporal buffer. The immediately following question is then if Theta corresponds to sensory data afferents, i.e. delivery of the current perceptual sensory state with accompanied reset of the network activity to the actually perceived location. Another question is how the temporal direction is reversed during the buffering, i.e. the change of direction of active place cells representing locations either before or after the current position. The loop-nesting is depicted in Figure 7.11b and the reversal issue is marked with a "?". A potentially suitable candidate to induce reversal of direction is assumed to be rebound spiking of neurons. Consider a network of neurons in which inverse directions of transitions inhibit each other. Given sufficient inhibition and vanishing activity in one direction, the inverse direction may respond by a post-inhibitory rebound spike. Rebound spiking was reported for principal neurons of the HF [10, 322], and, besides Theta, already suggested to have an impact on the formation of grid cells [141].

The effect of continuous temporal buffering of past and future places corresponds to in- and outbound traveling waves of activity in the Hippocampus, already described in [221]. It can be observed in the retrieval example of the temporal transition model, depicted in Figure 7.4. Thereby, it is believed that temporal buffering not only allows formation of grid scales. It is suggested that it also provides a continued prediction of possible future locations. It is likely that sweeping over potential traces is helpful for decision processes during goal-directed navigation, e.g. to allow propagation of reward values by changing synaptic efficacy by STP.

In the presented scale-space model, learning of multiple grid scales depends on Theta and Theta phase precession. A similar observation was reported for real grid cells in the rodent [189]. However, the smallest scale does not require Theta

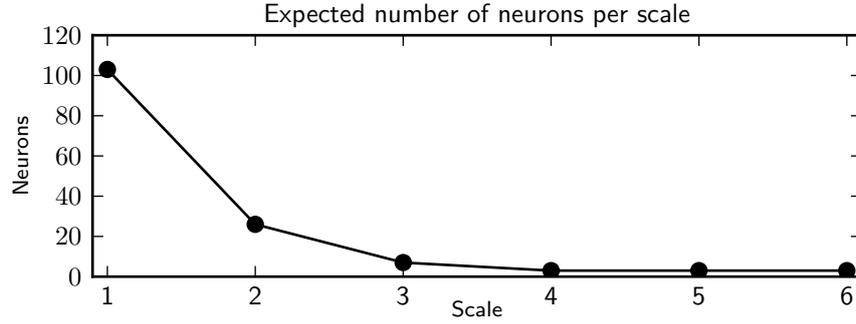
and can operate independently of both effects. Furthermore, there may exist other mechanisms besides Theta and Theta phase precession which provide temporally buffered data to form multiple scales. Nevertheless, the number of place cells active during one Theta cycle is reported to be limited as mentioned above [328], which is also reflected in the numbers of loops which can be nested within one Theta *main loop*. Without additional mechanisms to access more locations, the number of cells in the temporal buffer limits the number of scales which can be learned to  $\leq 10$ . Given that Theta is reported to oscillate in a range from 4 – 10 Hz [221] and the results depicted in Figure 7.9 which show that large scales lead to only little computational improvements, it is expected that around 5 to 7 grid scales will be found on average.

### 7.3.2 On the expected number of neurons per scale

Certainly, a non-artificial neuron is unable to associate with arbitrarily many inputs. Any neuron has only a finite number of synapses or dendrites available for local dendritic computation due to limitations of physical size. Given approximate numbers of the synapses,  $\sigma_{\text{eigen}}$  can be used to assess the number of neurons required to encode all expected transitions which appear on a certain scale. Note that the values used in the calculation in this section are conservative estimates if not stated otherwise, because absolute data is currently unavailable. Some assumption are backed by observations but re-interpreted in light of transition coding as soon as they are made.

Recordings suggest that  $\sigma_{\text{eigen}}$  on the smallest scale is between 10 to 20 cm [336], hence  $\sigma_{\text{eigen}} = 15\text{ cm}$  is used to compute the following trend on the number of neurons. Furthermore, grid cells are assumed to be re-used in contextually different areas. For instance, one room may differ entirely in its geometrical shape from another, thus one cell could in principle associate with one specific set of inputs in one room and another set in the other room without violating coherency and consistency constraints. However it is likely that cells tend to stick to previously learned association. The latter interpretation can be made on basis of recordings published by Derdikman et al. [85] who presented research in which grid cells of a rat that has to move in a hairpin maze tend to fire at the same location in every other corridor. The similarity of the firing locations is the head direction and the observable geometrical layout. Nevertheless it is suggested that the representational capabilities of the grid cell network are limited. For the sake of simplicity and understanding the trend of required neurons, the following computations are reduced for areas of the size  $10 \times 10\text{m}^2$ . In other words, the calculations are based on the assumption that any area of the given size could be considered contextually different than the next area of the same size. Certainly this value is fictional, as no data is available, but suffices for the sake of argument. Furthermore, interactions across scales could disallow some combinations and thereby generate sufficient context to discriminate between locations across scales. A combinatorial perspective on grid cell firing as has been presented previously by Mathis et al. [233]. The authors showed that certain combinations of states across scales should be considered invalid. In addition to these considerations, other cues, e.g. distal landmarks, could be used for contextual disambiguation of different *virtual* areas.

The number of synapses or dendrites per neuron differs significantly in literature. Values in the range from hundreds to thousands of synapses per neuron are stated. A conservative guess of  $N_{\text{syn}} = 2500$  synapses available for pre-synaptic connections



**Figure 7.12 – Expected number of grid cells per scale.** The number is computed according to Equation (7.2) with  $N_d = 250$  and the objective to cover the expected number of transitions in an area of  $10 \times 10 \text{ m}^2$ .

therefore does not seem devious. However, it is assumed that a single synapse is not sufficient to drive a grid cell. On the contrary, it is suggested that several synapses stimulated by different spatially modulated pre-synaptic neurons are required for a single cell to reach its spiking level. The number of necessary synapses used here is 50, which leads to  $N_d = N_{\text{syn}}/50 = 250$  dendrites to associate to pre-synaptic states. In the optimal case, a single grid cell however only associates with  $1/3$  of the pre-synaptic input space after self-organization to avoid violation of the *coherency constraint*.

The theoretically minimal number of neurons required to cover space is 3 as presented in Chapter 5. However, this only holds if the receptive fields of neurons either overlap by a small fraction or form Voronoi cells, thereby sampling the entire input space. In case of non-overlapping receptive fields, this number should be slightly but not significantly larger, and hence will be ignored in the assessment. Given the considerations of above, the trend for the total number of neurons required on any scale  $s \in 1, \dots$  in two dimensions follows according to

$$N_s = \lceil \max \left( \frac{3}{N_d} \frac{xy}{r_s^2} \frac{2}{3\sqrt{3}}, 3 \right) \rceil, \quad (7.2)$$

where  $r_s = 2^{s-1} \sigma_{\text{eigen}}$ , and  $\lceil \cdot \rceil$  is the ceiling function. The first fraction accounts for covering one third of the input space with the available dendrites, whereas the remaining two fractions give the densest packing of circles for a rectangular arena of size  $x \times y$ . The max operator is used to prevent violation of the coherency constraint.

Figure Figure 7.12 visualizes the trend for the estimated numbers. The trend follows an exponential decay of the number of neurons required per scale, and levels at the bare minimum of neurons already on the fourth scale. The proposed scale-space model is therefore in agreement with data reported by Stensola et al. [336], who found only few grid cells corresponding to larger scales.

It is expected that other factors, for instance the necessity for redundant representations, particular tunings of grid cells towards certain cues, or the behavioral necessity to over-represent one or more scales, have an impact on the exact numbers. However, it is suggested that the general trend of an exponential decay of the number of neurons for increased scales remains, given sufficiently many recordings.

### 7.3.3 Relationship to algorithms and concepts from computer science

Recall the graph theoretical interpretation of temporal and spatial transitions as well as place cells. In this notation, spatial symbols correspond to vertices in a graph. Spatial and temporal transitions form edges between vertices. In light of this notion, the presented algorithms correspond to parallel variants of Dijkstra's algorithm and A\* [87, 137, 138]. Both are simple yet powerful and therefore widely-used and adopted algorithms. Dijkstra's algorithm computes a *minimal cost traversal* of a graph, i.e. it finds the shortest route from a given vertex to a target vertex while considering costs that are assigned to edges of the graph [87]. In terms of spatial navigation, this corresponds to navigating from a start to a target location. Edge weights correspond to cost-of-travel, likely represented in form of reward signals. A\* extends Dijkstra's algorithm by including heuristics to improve computational times and was developed for spatial navigation on robots [137, 138]. Multiple scales for transition encoding and end detection could be considered to represent heuristic information.

In computer vision, scale-space representations are often used in combination with feature descriptors such as Scale-Invariant Feature Transform (SIFT) or Speeded-Up Robust Features (SURF) [18, 219, 220]. After searching for points-of-interest in image representations on several scales, the points are described in terms of the descriptors which exhibit a compact representation of the point and its surround. Typically, the feature descriptors allow comparison and are used to find features across images. The result are algorithms with capability to search and track features in real-time. It is therefore likely that a similar approach, namely extraction of scale-space representations of sensory data, yields spatial feature descriptors. It is further believed that such a spatial feature descriptor can be used for spatial navigation in robotics scenarios.

As mentioned above, the separation of spatial transitions (grid cells) and spatial symbols (place cells) can be understood as an abstraction layer, for instance the *Bridge*, *Facade*, or *Mediator* design patterns [119], as follows. Knowledge of spatial neighborhoods is stored in grid cells instead of direct binding this information to place cells. Thereby, spatial sequences can be retrieved without reconstruction of the sensory states that lead to the sequences. Furthermore, the sensory representation may change over time while the spatial neighborhood information remains. On the other hand, spatial information can change while grid cell representations can be re-used. The abstraction thereby provides a powerful mechanism to separate details, and increases re-usability and fault-tolerance.

## 7.4 Conclusion and future work

The scale-space model presents an entirely novel interpretation of the entorhinal-hippocampal loop. Just as most other models and recordings it is proposed that place cells store spatial locations [131, 264]. However and in contrast to any existing model, grid cells are suggested to optimally encode spatial transitions in a scale-space representation. The scale-space increment from one scale to the next is ideal at a value  $\sqrt{2}$ , which is the result of a look-ahead operation combining multiple transitions. The look-ahead operation itself is required to improve the computational performance of the suggested transition model to make it behaviorally relevant.

At the moment, the proposed scale-space model has no measure of quality assigned to transitions. Hence it yields only the availability of a trajectory without returning the most likely or best according to some measure. Nevertheless it demonstrates feasibility of the approach. Adding a quality assessment to trajectories with the help of a reinforcement signal is certainly possible. In fact, this was proposed already by Cuperlier et al. in a biologically plausible model that stores rewards alongside transitions in a spatial navigation task [73, 74, 146]. Therefore, future work will include RL as a selection mechanism of available trajectories.

However, it has to be determined where and how the reward is stored and how it can be retrieved. Two candidates are likely. The first is persistent modulation of the synaptic weights due to LTP [250]. This technique was successfully used in the model by Cuperlier et al. [73, 74, 146], but limits re-usability of place and transition cells. The second candidate is STP which modulates the synaptic efficacy in an abrupt but non-lasting manner and was reported to exist in hippocampal neurons [52, 308]. STP is presumably able to tune the neural response times such that preferred transitions will spike early. In combination with sufficiently fast inhibitory interneurons such as discovered by Diba et al. in hippocampal neurons [86], this will likely block any alternative route or unwanted trajectories after just few iterations. The result should resemble the activity reported by Pfeiffer et al. [282]. A novel model addressing this issue by STP as well as Theta phase precession and rebound spiking is currently in development. Furthermore, this novel model will address the simplifications and discussed numerical issues. It will prove useful to investigate if the results are in fact similar to the findings by Pfeiffer et al. [282] or not. Furthermore, it will be essential in understanding multi-homing tasks in which a simulated agent has to select either of many target locations according to the reward that it will receive.

A transition model with a trajectory selection mechanism based on reward modulation will allow to study its behavioral impact. It is believed that the separation of temporal and spatial transitions as proposed in the scale-space model will lead to very specific observations. For instance, consider the Morris water maze experiment in which an animal has to locate a platform which is submerged in an opaque liquid [249]. It is predicted that an animal will follow the first trajectory to the target that it found in the absence of a spatial transition system but presence of a temporal transition system during learning. In contrast, results of experiments in which a specific temporal order is required to reach a certain goal location will be impoverished without temporal transition encodings.

To summarize, the scale-scale model allows an interpretation of the entorhinal-hippocampal loop in terms of an optimal storage device for transitions. Thus future work and extensions of the model will focus on re-examination of recordings in light of this interpretation.



**Part III**  
**Beyond MTT**  
**and**  
**Neural Spatial Navigation**



## Chapter 8

# Towards massively distributed spatial navigation

In contrast to spatial navigation in the brain, many researchers argue that basic spatial navigation for robots is a solved issue and future research should focus on the robustness of the employed techniques [49]. Decades of research brought forward many successful solutions to Simultaneous Localization and Mapping (SLAM) and related issues like image perception and understanding. For instance, approaches rooted in probabilistic techniques such as particle filters, advocated especially by Sebastian Thrun, were employed to robot localization, self-driving cars, and found widespread application in several other areas [348–350, 353].

Other technical solutions have stronger resemblance with biological findings [61, 110, 296, 378]. Here, topological spaces of connected locations are formed instead of dense maps of the surrounding. A topological map itself is sufficient to help a robot find its way to target locations [378]. Interestingly, it was suggested that the activity of the place cell network in the rodent Hippocampus indeed forms a topological map [75, 76, 149]. This property was previously observed and exploited in [77], where a robotic model for navigation which forms a topological map of place cells was used.

The theory presented in the previous part of the thesis is clearly in favor of the topological map assumption. Grid cells are proposed to provide metric information linking one location to another. Thus, the stored items and transitions within the entire scale-space transition system form a topological space.

Topological and graph theoretical notations have several benefits. For instance, there is no necessity to store a dense map of the environment, which typically requires tremendous amounts of memory for very large environments. In contrast to dense maps, only important locations and their spatial description, e.g. in form of sensory readouts such as boundary or geometric information, have to be stored alongside the transitions from one location to another. This principle was previously used in robotics scenarios, e.g. by Franz et al. already in 1998 [110], and extended to multi-robot mapping [153, 154].

Why is multi-robot mapping relevant? Robots need to interact not only with the environment. In addition, they are required to cooperate with each other and their human operators. Thus they are required to share information by communication. However, knowledge about how to distribute data properly in a swarm of cooperating robots is limited [49]. Spatial navigation, and in this context the representation of the environment in form of a topological map, was therefore chosen as fundamental

example for the distribution of data. It is believed that it leads to generalizable data structures and algorithms.

The focus of the work presented in this chapter was not to develop methods that run faster than state-of-the-art techniques. Contrarily, the focus was to analyze and characterize the suitability of algorithms and data structures for global distribution and local parallelization on large numbers of participating robots, or, in other words, in a massively distributed setting.

The results presented in this chapter were conceived collaboratively with *Edvarts Berzs*, whom I supervised during his Master's thesis. The results were submitted for peer review [372], and were partially presented already in his Master's thesis [22]. Therefore, portions of [372] and [22] are reprinted verbatim with permission. In particular, definitions and mathematical formulations are reprinted *as-is*.

## 8.1 Problem formulation and related work

Most techniques for SLAM focus on navigation and mapping for a single robot. Albeit there are suggestions for multi-robot mapping, e.g. suggested by Thrun et al. [351], there is no convergence perceivable towards a single or just a few prominent techniques in the community. One of the difficulties in multi-robot mapping is the distribution of the map. Even though multiple robots can cooperatively map an environment [327,371], there is often the necessity to store a map at a central location for path planning processes. Recent surveys on the future state of multi-robotics research therefore point out that, besides many other problems, further investigations are required to address the issue of a truly distributed map on all participating agents [127,307].

Autonomous agents that operate in unmapped environments typically accumulate significant amounts of data to solve SLAM [90,329,352]. When extended to multiple robots [154,351], the issue also involves finding a suitable method which allows access to an entire map for all participating agents. One solution is to store the map at a central location [371]. However this comes at the cost of a single source of failure.

Consequently, approaches are required in which the map can be distributed across participating hosts [49]. On the one hand, suitable data structures have to be defined which allow simple distribution and express fault tolerance. On the other hand, distributed maps are prone to decrease of computational performance with respect to retrieval times. For localized data, several contributions improved the runtime and memory complexity of the involved algorithms significantly during the last decades [1,16,94,152,170,171].

To address the issue of distributability, data structures and algorithms were defined which were inspired by the results presented in the previous chapters and on findings reported in literature. In particular, data in the form of a graph is distributed with the help of two multi-layer data structures. Retrieval of shortest distances and paths are based on Dijkstra's algorithm, the layers are constructed by a contraction process in which vertices of the input graph are pruned.

Dijkstra's algorithm is one the most fundamental algorithms in computer science [87], and its currently optimal reported runtime of  $\mathcal{O}(E + V \log V)$  in a graph with  $V$  vertices and  $E$  edges is achieved with the help of preprocessing in Fibonacci heaps. However, Fibonacci heaps were considered unsuitable for distributing data

which is accumulated over time in the work presented here. Different other methods were suggested to speed up graph preprocessing. The most successful techniques use hierarchical representations of the input data and generate heuristic information [150, 184, 234, 318]. Other sophisticated hierarchical clustering methods to speed up shortest path computations include, but are not limited to, Hierarchical Encoded Path Views (HEPV) [155, 156]), Hierarchical Performance Multi-Level Routing (HiTi) [175], multi-level graphs such as in [17, 234], or recursive trees as in [397]. Typically, the recursively constructed hierarchical methods use specialized partitionings of the input graph [38, 82, 83].

## 8.2 Algorithms and Data Structures

The two data structures, termed Sparse Layered Graph (SLG) and Transition Graph (TG), will be introduced formally in this section. The notation is inspired by previous work by Jing et al. [170], Schulz et al. [318], and Delling et al. [83]. Note that  $|\cdot|$  denotes cardinality of a set, subscripts denote layer indexes, and elements of a layer are denoted by superscripts. For instance,  $v_i^j \in V$  is the  $j$ -th element of  $V$  at the  $i$ -th layer.

First, definitions for a single layer are presented in which indexes for layers are omitted. Then, the definitions are extended to multiple layers. Subsequently, the algorithms for the construction and retrieval are described.

### 8.2.1 Definitions for single and multiple layers

The construction of an SLG and TG requires a graph as input which contains nodes that are associated with coordinates. Edges of the graph are associated with weights, and correspond, for instance, to time-to-travel [318]. The graph is defined as follows, using standard graph theoretical notation. Note that the terms *vertex* and *node* are used synonymously.

**Definition 6.** Let  $G = (V, E)$  be an undirected planar graph, where  $V$  is the set of vertices and  $E$  the set of edges  $E = \{e(u, v) : \text{there exists an edge between } u, v \in V\}$ . Each edge  $e(u, v) \in E$  is associated with a weight denoted as  $w(u, v) \geq 0$ . Each node  $v \in V$  is associated with a coordinate  $x(v) \in \mathbb{R}^2$ , where all coordinates  $x(v)$  are assumed to be uniformly distributed in the coordinate space  $\mathbb{R}^2$ .

An SLG consists of several clusters  $c_j$  which group the input data. Thus, each vertex of the graph will be assigned to a cluster  $c_j$  based on its coordinate. However, clusters are in principle not required to form a partitioning of the input space.

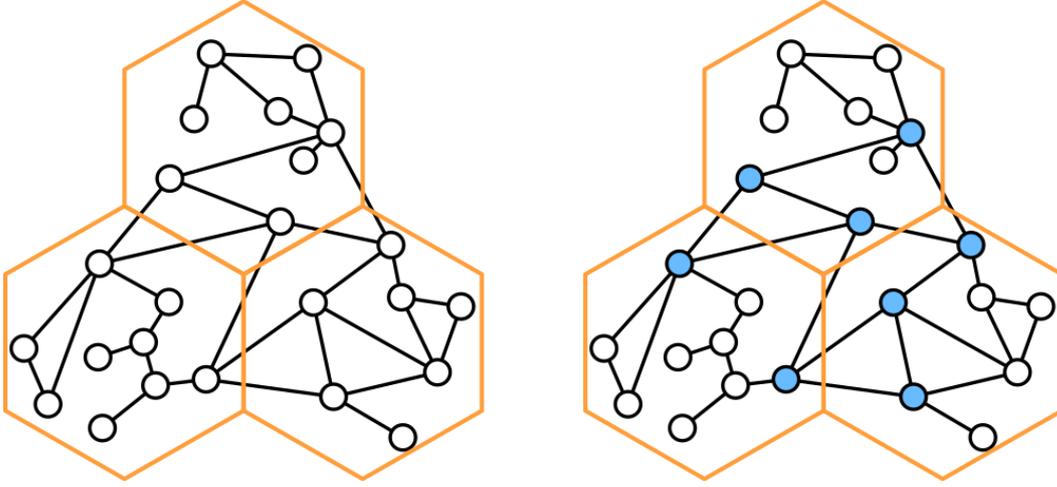
**Definition 7.** Let  $G = (V, E)$  be a graph. The covering  $C$  of  $G$  is the collection of subsets

$$C = \{c^j \subseteq V : j \in J\},$$

such that

$$V = \bigcup_{j \in J} c^j,$$

where  $J = \{1, 2, \dots, N\}$  is an index set and  $N = |C|$ . Each subset  $c^j$  is accompanied by a coordinate  $\zeta^j = \zeta(c^j) \in \mathbb{R}^2$ , called cell center.



**Figure 8.1 – Graph, cells, and border nodes.** An input graph (left hand side, small circles and black lines) is covered by a set of cluster cells (orange hexagons). All border nodes from these cells are extracted for further use in the Transition Graph (TG) (right hand side, filled small circles). (Figures reprinted with permission from [22, 372])

Note that  $c^j \cap c^k = \emptyset, \forall c^j, c^k \in C$  is not guaranteed and that the terms *cluster* and *cell* are used exchangeably. In the work presented here, the cluster centers are distributed hexagonally and clusters form a partition of the space. However, irregular distribution of cluster centers and coverings are feasible with the described methods.

The second data structure, TG, contains all nodes  $v \in B$  that have edges which cross cell borders. The set of border nodes  $B$  is defined formally as follows and illustrated in Figure 8.1.

**Definition 8.** The border node set  $B = \{v^b : v^b \in V\} \subseteq V$  is the set of nodes  $u^b, v^b$  for which

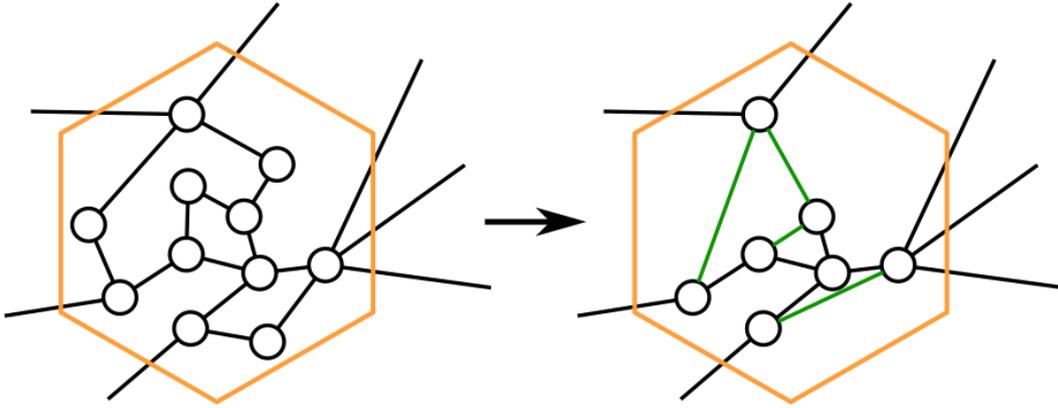
$$\exists j, k \in J, j \neq k, \exists e(u^b, v^b) \in E : (u^b \in c^j) \wedge (v^b \in c^k)$$

holds.  $B^c = \{v^b : v^b \in c\}$  is the border node set of cell  $c$ . The set of all edges between pairs of border nodes is defined as

$$F = \{e(u, v) : u, v \in B \wedge e(u, v) \in E\}.$$

The TG can be used to accelerate computations of larger trajectories. After computation of any all-pairs-shortest path within each cell of a TG and *locally buffering* the information, long distance routes can be computed approximately by TG.

Each of the above definitions can be extended to multiple layers. Given a number of layers  $L$  with strict total order  $I = (0, \dots, i, \dots, L - 1)$  then  $G_i = (V_i, E_i)$  denotes graph  $G_i$  of the  $i$ -th layer with vertex set  $V_i$  and edge set  $E_i$ . Furthermore, the SLG and TG of layer  $i$  are said to contain node and edge sets  $(V_i^S, E_i^S)$  and  $(V_i^T, E_i^T)$ , respectively. Note that intermediate indexes are dropped if they are clear from context, e.g.  $B^{c_j} := B^j$ .



**Figure 8.2 – Edge contraction during the construction of a Sparse Layered Graph (SLG).** The contraction process removes nodes with minimum degree within each cell and inserts new edges between the remaining nodes (indicated in green) where necessary. As a consequence the number of nodes in the layer is reduced, whereas the number of edges may increase depending on the structure of the input graph  $\mathcal{G}$ . (Figure reprinted with permission from [372]).

### 8.2.2 Construction algorithms for the data structures

The entire SLG  $\mathbf{S}$  consists of  $L$  consecutively numbered layers, i.e.  $(0, \dots, L - 1)$ . In turn, each layer  $\mathbf{S}_i$  consists of a covering  $C_i$ . Formally, this is defined as follows.

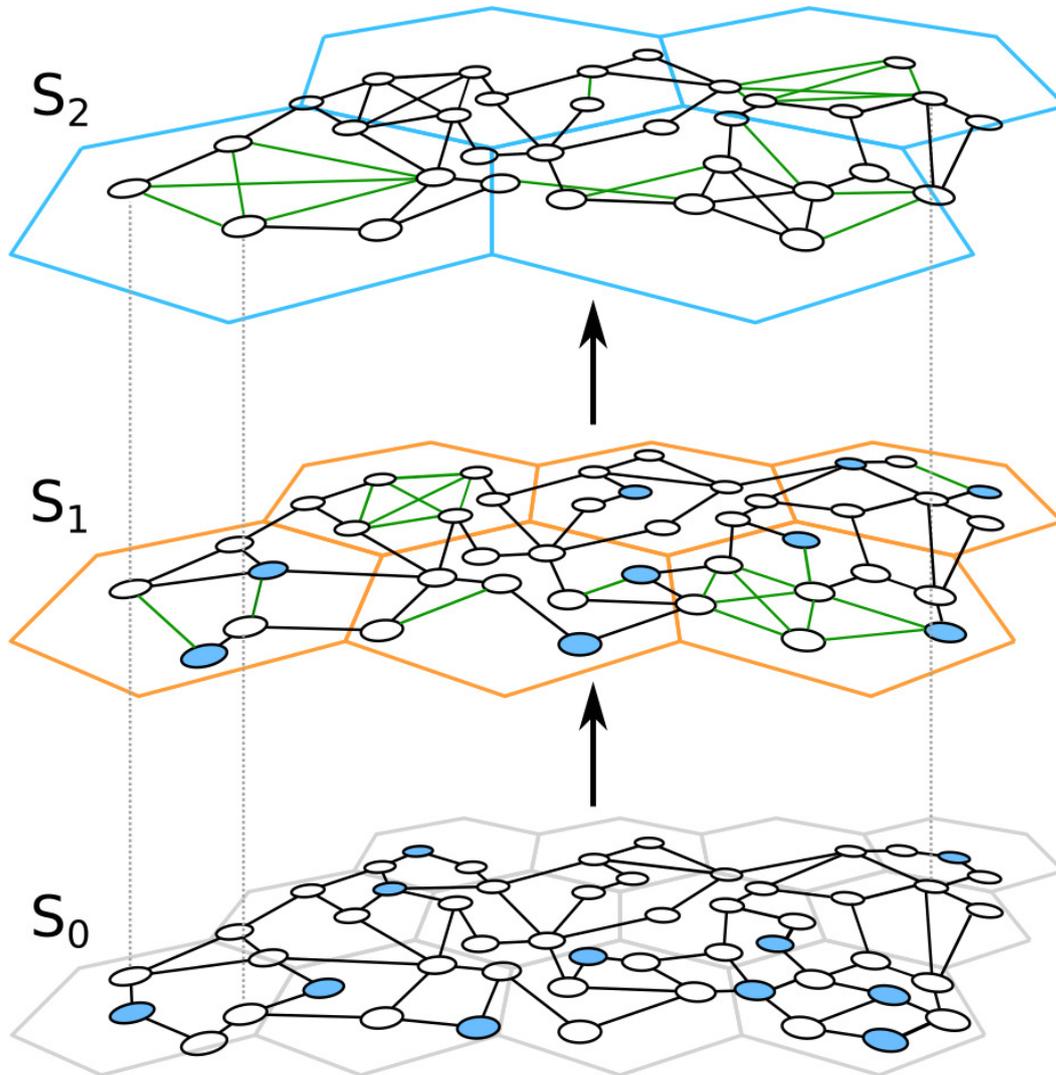
**Definition 9.** An SLG  $\mathbf{S}$  is the union of all of its layers, i.e.

$$\mathbf{S} = \bigcup_{i \in I} \mathbf{S}_i, \quad \mathbf{S}_i = \bigcup_{j \in J} \mathbf{S}_i^j,$$

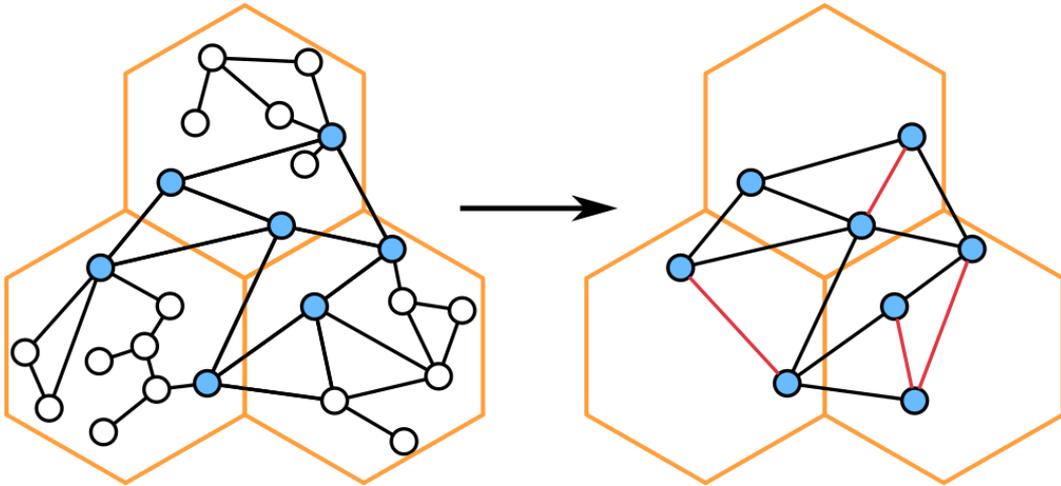
where  $\mathbf{S}_i$  represents the SLG of layer  $i$ , which itself is a cover of all subgraphs induced by the covering  $C_i$ . Consequently,  $\mathbf{S}_i^j := \mathbf{S}_i^{c_i^j}$  is the subgraph induced by cell  $c_i^j$ .

According to Definition 7,  $\mathbf{S}_0$  consists of a covering of the original graph  $\mathcal{G}$ . Each consecutive layer  $j = 1, \dots, L - 1$  is formed by pruning the nodes of the previous layer during a *node contraction* process. All remaining nodes are re-assigned to clusters on layer  $j$ . During node contraction, nodes with a modifiable minimal degree are removed. Novel edges are introduced for all *dangling nodes*, i.e. all vertices which had an edge to the removed node on layer  $j - 1$  are now directly connected on layer  $j$ . Note that the contraction of nodes within a single cell can be performed independently of and thereby in parallel to other cells of the current layer. Newly introduced edges are tagged as *contraction edges*. By construction, each consecutive layer will have *at most* as many nodes as the lower layers, and in most cases strictly less nodes than previous layers. However, the number of edges may increase in the worst case. Contraction in one cell is visualization in Figure 8.2. An example of an SLG with three layers and its construction is depicted in Figure 8.3. Pseudocode for the algorithm is provided in Algorithm 1 in Appendix D.

Once layer  $\mathbf{S}_i$  is generated, the corresponding TG  $\mathbf{T}_i$  can be constructed by extracting the set of border nodes on the respective layer. Formally,  $\mathbf{T} = \bigcup_{i \in I} \mathbf{T}_i$ , where each  $\mathbf{T}_i$  consists of the TGs of all cells  $c_i^j$ , i.e.  $\mathbf{T}_i = \bigcup_{j \in J} \mathbf{T}_i^j$ . Afterwards, all-pair-shortest paths are computed for all border nodes in each cell of TG  $\mathbf{T}_i$ .



**Figure 8.3 – Construction of a Sparse Layered Graph (SLG) with multiple layers.** The SLG  $S$  consists of several layers  $S_i$  (three layers shown here), each layer inducing its own subgraph. Additionally, each layer  $S_i$  consists of cells  $c_i^j$  with their own subgraph  $S_i^j$ . The first (bottom) layer is formed by covering the input Graph  $G$  with a predefined number of cells (gray hexagons, bottom row). Every following layer is constructed by pruning the set of nodes by a *contraction process*. Every next layer (top row) receives the remaining nodes and edges as input and operates on cells with a fixed size increment (blue hexagons). (Figure reprinted with permission from [372]).



**Figure 8.4 – Construction of a Transign Graph (TG).** The TG for the graph presented in Figure 8.1 (left hand side) is constructed by extracting all border nodes of each cell, and adding the all-pair shortest path information within each cell (red lines on right hand side) if necessary. (Figure reprinted with permission from [22, 372]).

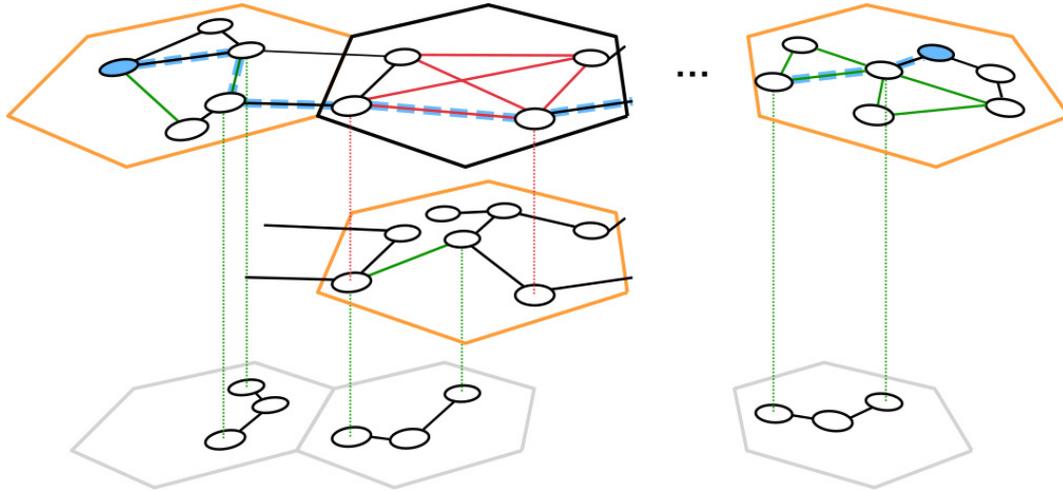
While the construction of SLG allows parallelizing node construction across cells, the formation of TG can only be parallelized in terms of layers. The algorithm is illustrated in Figure 8.4.

The construction algorithms were analyzed with respect to their run-time complexity. With a suitable data structure to store and access elements of the graphs, such as a hash table, the upper bound on a single layer of an SLG is  $\mathcal{O}(E^2)$  and without such a data structure  $\mathcal{O}(E^3)$  in the worst case. In an expected average case, the bound drops to  $\mathcal{O}(V \cdot \log V + V)$ . Considering multiple layers, the bounds are governed by the number of edges and reside in  $\mathcal{O}(E^2)$  as long as the number of layers  $L \ll E$ . Analysis of the construction of TG in multiple layers yields a worst case of  $\mathcal{O}(\sum_{i=0}^{L-1} B_i (E_i^S + V_i^S \log \frac{V_i^S}{C_i}))$  and an expected average case of  $\mathcal{O}(L \cdot (E + V \log V))$ . Here,  $C_i$  is the number of cells of the layer and  $E$  and  $V$  are maximal numbers of vertices and edges in the entire graph. For details of the derivation, see Appendix E.

### 8.2.3 Algorithms for retrieval

Retrieval of shortest distances and routes are based on Dijkstra's algorithm [87]. Computing the shortest distance from a source node  $s$  to target node  $t$  can be performed mostly in TG  $\mathbf{T}$ . First the lowest level in SLG  $\mathbf{S}$  in which both the source  $s$  and target  $t$  are present, i.e. not contracted, is determined. Afterwards the corresponding cells of  $\mathbf{S}$  in which  $s$  and  $t$  reside are merged with  $\mathbf{T}$  on the respective layer. Finally, the shortest distance can be computed on the merged graph. Obviously, searching in layer  $k$  is unlikely to be optimal in all cases. For instance, if both nodes are contracted on the lowest layer, all operations will take place on this layer. Optimization of such cases and finding a technique to efficiently propagate retrieval to the highest layer is left for a future study, though.

Retrieval of a shortest path requires an additional step when compared to extracting the shortest distance. The shortest distance query already yields a consecutive



**Figure 8.5 – Example for a shortest path query in SLG and TG.** The goal of the operation is to determine the shortest path, indicated by the dashed blue line between the two blue nodes in the topmost row. First, the highest level on which both vertices are not contracted is identified and their respective cells (top row, orange) of the SLG are combined with the TG to find the shortest path using only intermediate TG cells (in black). Then, all-pair shortest path edges (red edge in top row) of the shortest path in the TG are resolved to their respective edges in SLG cells (middle row). Finally, each contracted edge (green edges on shortest path) is resolved using lower level representations (bottom row). This process is invoked recursively and in parallel until all remaining contracted edges are resolved. (Figure reprinted from [372]).

path which *necessarily* contains only vertices on the shortest path. However, it may contain several *contraction edges* as well as edges which were introduced during the all-pair-shortest-path construction of the TG. All such edges can be resolved by querying cells of lower layers in parallel. The shortest path query is depicted in Figure 8.5 and Pseudocode for the algorithms is provided in Algorithm 3 and Algorithm 4 (see Appendix D). The figure shows that, as soon as a contracted or an all-pairs-shortest edge is detected, the retrieval is propagated downwards in the hierarchy until the entire path is resolved on the smallest layer. Complexity analysis of the retrieval of a shortest path revealed a worst-case run-time of  $\mathcal{O}(E_k^{\mathbf{S}} + V_k^{\mathbf{S}} \log V_k^{\mathbf{S}})$  (see Appendix E for details).

### 8.3 Discussion and future work

The introduced algorithms allow distribution of graphs, and computation of shortest paths on the distributed graph. The complexity analysis revealed that the approach to construct the data structures is suitable for real-world scenarios. However, the time complexity of the query operations is far from ideal and will require further investigations. Simulations performed by Edvarts Berzs showed the feasibility of the construction methods [22]. Furthermore, the theoretical finding that the algorithms for retrieval have to be improved significantly was also supported by simulations. Recall however, that the focus of the algorithms was not an improvement on the run-time, but on a distribution technique of the data structure to multiple hosts.

So far, the algorithms were only analyzed in the case of sequential execution.

Nevertheless, the distribution of data according to the proposed data structures shares similarities with existing *peer-to-peer* network technologies. In these technologies, one of the main issues is addressing remote content. It is often solved by distributed hash tables, e.g. [396], which is related to clustering data in multiple layers as performed in the proposed technique. Furthermore, resolving contracted edges to edges of the original graph corresponds to routing packages in networks to a target host. Thus, it is believed that the data structures can be used as an overlay on existing fully decentralized network topologies.

Given the presented results, future work has to focus on improved algorithms for the query operations. Additionally, the construction of the TG will be revised as it cannot be performed in parallel for each cell at the moment. A distributed variant of the proposed data structures is currently in development.



## Chapter 9

# Concluding remarks and potential directions

The thesis introduced several novel concepts and perspectives. The *bundling trick* of MTT was developed to analyze the optimization problem of finding an ideal representation of transitions with a minimal number of bundles. As a consequence, the optimal placement of transition encoders in two dimensional space was found to be a hexagonal arrangement.

Given the assumption of distributed and independent dendritic computations, it was further possible to derive the error function of a grid cell. It was demonstrated that such a cell converges to a hexagonal arrangement of its response fields with only simple terms in the error function. Furthermore, the error function allowed to derive a competitive network of grid cells. The main reason for a purely inhibitory network were the mandatory constraints given by MTT and MTS, namely axioms to generate coherent and consistent trajectories, showing that neural computations can be derived from their algorithmic understanding and their computational purpose.

After demonstrating that a network of competitive grid cells cover the input space during simulations, their algorithmic interactions with place cells were examined. The computational point of view lead to the observation that the network in only a single scale has a behavioral issue, namely the runtime of the path planning operation. Introduction of a scale-space representation of transitions, encoded by multi-scale grid cells, lead to an exponential speed-up of the computation and thereby solved the problem. The model exposes several requirements and consequences which can be related to biological observations such as buffering, an ideal increment of scales at a factor of  $\sqrt{2}$ , or a distinction between temporal and spatial transitions to name just a few. Temporal buffering of information, for instance, is necessary to learn transitions in the scale-space representation, which itself is assumed to be expressed in form of Theta phase precession. Ultimately, the model lead to the description of a novel perspective of the entorhinal-hippocampal loop and the recurrent interactions of spatially modulated afferents, place, and grid cells as a storage device for spatio-temporal transitions which can perform look-ahead to accelerate computations.

Finally, the concepts of local clustering and transitions were applied to a technical system. The proposed novel data structures for the computation of navigational trajectories in a swarm of robots resemble concepts of peer-to-peer networks, well established and widespread applied in computer science.

Given these results, several directions of future research are possible which are briefly outlined as follows. The biologically plausible model of competitive grid cells could be refined using spiking behavior and elaborate neural models. The thereby

modified model could lead to novel insights about the required temporal dynamics during plasticity, and the self-organization of receptive fields during learning of transitions. In addition, detailed modelling could lead to predictions with respect to local synaptic circuits. Another possible direction is to investigate self-organizing principles in the scale-space model. Currently, the model has pre-defined scale increments after they were derived mathematically. A self-tuning process to organize the scales appears to be more likely though. One domain that could provide hints on how to describe such a model is information theory, which has been used recently to prove that modularization and the concept of minimum description length yields optimal solutions in the quest for efficient neural codes [240]. It is proposed that a similar approach and understanding rooted in concepts of information theory and optimal coding theory could lead to a descriptive self-organizing scale-space model of grid cells. Furthermore, the currently existing abstractions and numerical simplifications that were introduced could be removed. In addition, the presented scale-space model used spatial coordinates to generate the pre-synaptic activity of both place and grid cells to avoid modelling of pre-synaptic sensory states. Work is already in progress to model the sensory input space to place and grid cells explicitly in form of boundary vector inputs both for the purpose of biological plausibility as well as for a robotics scenario. Finally, the technical application of the concepts should be extended and re-examined. The available results for the massively distributed robotic scenario of path planning are not overwhelming yet. However, future research could focus on including the most recent developments of high performance distributed computing and peer-to-peer systems to address the existing performance bottlenecks during query operations. Consequently, the data structures should be adapted in an appropriate way.

Entirely novel directions of research and applications of MTT to new domains are equally possible. For instance consider grasping with a robotic arm which requires the movement of an arm segment along a specific trajectory. MTT and a spatio-temporal MTS in particular allow to phrase the problem in terms of local neighborhood relationships of joint-states of the robotic arm. Using MTS to encode the states and intermediate transitions has the benefit of removing a global coordinate system, which is commonly used in this task. A scale-space representation is thought to allow the robot to find shortcuts in the arm movements by itself. It is further believed that the MTS can easily represent and relate noisy sensor measurements, which typically occur in real-world robotics scenarios.

The novel theory itself as well as the formalisms to describe MTS were deliberately kept as abstract as possible. It is proposed that the technique is capable to account for other neurons in the mammalian brain which have to encode transitions, not only grid cells. Grasping an object, or the production of vocal sequences, are just two of many examples in which sequences and transitions between intermediate symbols are relevant.

To conclude, this thesis introduced MTT and applied it to the problem of spatial navigation in the rodent brain. In turn, this led to the development of an entirely novel model for competitive grid cells which encode transitions in form of a scale-space representation. It was possible to relate recordings and observations from real rodents to consequences of algorithmic interactions within the model and optimality considerations. Finally, the thesis transferred the novel concepts to technical systems.

# Appendices



## Appendix A

### Partial derivatives of the error function of the single grid cell model

The partial derivative of  $L(\mathbf{w})$  with respect to  $w_k$  is

$$\frac{\partial}{\partial w_k} L(\mathbf{w}) = -\frac{2}{N}(1 - w_k). \quad (\text{A.1})$$

When omitting the normalization constant  $\frac{1}{N}$ , the partial derivative of  $E(w)$  is given by

$$\frac{\partial}{\partial w_k} E^+(\mathbf{w}) = \frac{\partial}{\partial w_k} \sum_i \sum_j A_{ij}^+ (w_i - w_j)^2 \quad (\text{A.2})$$

$$\begin{aligned} &= \overbrace{\frac{\partial}{\partial w_k} \sum_{i \neq k} \sum_{j \neq k} A_{ij}^+ (w_i - w_j)^2}^{=0} + \overbrace{\frac{\partial}{\partial w_k} A_{kk}^+ (w_k - w_k)^2}^{=0} \\ &\quad + \frac{\partial}{\partial w_k} \sum_{j \neq k} A_{kj}^+ (w_k - w_j)^2 + \frac{\partial}{\partial w_k} \sum_{i \neq k} A_{ik}^+ (w_i - w_k)^2 \end{aligned} \quad (\text{A.3})$$

$$= 2 \sum_{j \neq k} A_{kj}^+ (w_k - w_j) - 2 \sum_{i \neq k} A_{ik}^+ (w_i - w_k) \quad (\text{A.4})$$

$$= 2 \left( \sum_{j \neq k} A_{kj}^+ w_k - \sum_{j \neq k} A_{kj}^+ w_j \right) - 2 \left( \sum_{i \neq k} A_{ik}^+ w_i - \sum_{i \neq k} A_{ik}^+ w_k \right) \quad (\text{A.5})$$

$$\begin{aligned} &= 2 \left( \sum_j A_{kj}^+ w_k - A_{kk}^+ w_k - \sum_j A_{kj} w_j + A_{kk}^+ w_k \right) \\ &\quad - 2 \left( \sum_i A_{ik}^+ w_i - A_{kk}^+ w_k - \sum_i A_{ik} w_k + A_{kk}^+ w_k \right) \end{aligned} \quad (\text{A.6})$$

$$= 2w_k \left( \sum_j A_{kj}^+ + \sum_i A_{ik}^+ \right) - 2 \left( \sum_j A_{kj}^+ w_j + \sum_i A_{ik}^+ w_i \right) \quad (\text{A.7})$$

$$= 2w_k \left( \sum_j A_{kj}^+ + \sum_i A_{ik}^+ \right) - 2 \sum_i w_i (A_{ki}^+ + A_{ik}^+) \quad (\text{A.8})$$

$$= 4w_k \sum_j A_{kj}^+ - 4 \sum_i w_i A_{ki}^+ = 4w_k - 4 \sum_i w_i A_{ki}^+, \quad (\text{A.9})$$

where the last line is due to the symmetry, i.e.  $A_{ki} = A_{ik}$ , and because  $\sum_j A_{kj}^+ = 1$ .  $E^-(\mathbf{w})$  follows accordingly.



## Appendix B

### Calculation of gridness score, grid orientation, and orientation error

The gridness scores were computed using the common procedure described by Sargolini et al. [313]. In contrast to Sargolini et al. [313], the auto-correlation was not computed on a smoothed rate-map of the neurons output but directly on the dendritic weight maps  $W_i$  for cell  $i$  as discussed in Chapter 6 though. The Pearson correlation  $R$  is defined as the two dimensional auto-correlation according to

$$R = \frac{1}{N s_{W,i}^2} \mathbf{E} \left[ (W_i - \mu_{W,i})^2 \right], \quad (\text{B.1})$$

and was evaluated for the two dimensional weight map  $W_i$  with. Here,  $\mathbf{E}[\cdot]$  denotes the expectation,  $N$  the number of entries in the (square) weight map,  $\mu_{W,i}$  is the mean value of  $W_i$  and  $s_{W,i}$  its standard deviation. In case of a grid cell with hexagonal firing fields, the hexagonal pattern will emerge in the autocorrelogram.

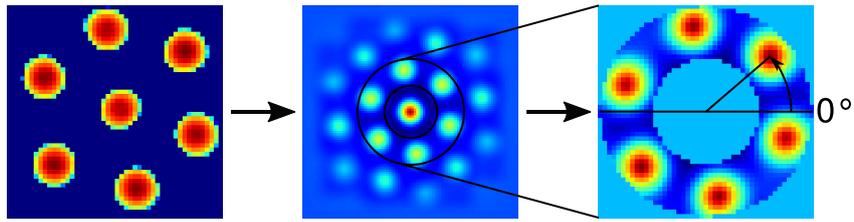
Subsequently, a ring in which the six peaks were expected was extracted from the auto-correlogram. The radius and width of the ring were determined by receptive field size parameter  $\sigma_1$  of the grid cell model. The inner ring and therefore central peak, which was not included in further analysis, was of size  $2N_{\text{dendrites}}\sigma_1$  and the outer ring which limits the extraction area was of size  $4N_{\text{dendrites}}\sigma_1$ .

The ring of extracted auto-correlated weights was afterwards correlated with rotated versions of itself. The correlation values  $c_d$  were computed for angles  $d$  in the range from  $0^\circ$  to  $180^\circ$  in steps of  $1^\circ$ . The gridness score  $g_{\text{score}}$  was finally computed as the difference between the lowest correlation value for a rotation of  $60^\circ$  or  $180^\circ$  and the maximal correlation value for the rotations of  $30^\circ$ ,  $90^\circ$ , or  $150^\circ$ , i.e.

$$g_{\text{score}} = \min(c_{60^\circ}, c_{120^\circ}) - \max(c_{30^\circ}, c_{90^\circ}, c_{150^\circ}). \quad (\text{B.2})$$

The grid orientation  $o_i$  of a cell  $i$  was computed by selecting the angle between the horizontal axis with origin in the center of the auto-correlogram and the closest maxima of the auto-correlogram in the counter-clockwise orientation, similar to the procedure described by Hafting et al. [135] or Sargolini et al. [313]. The resulting angle is thereby in the range  $[0, 60]^\circ$ . The average orientation error  $e_{\text{avg}}$  between  $N$  cells was computed as

$$e_{\text{rel}} = \frac{1}{N} \sum_i \sum_{i \neq j} |o_i - o_j|. \quad (\text{B.3})$$

**B. Calculation of gridness score, grid orientation, and orientation error**

**Figure B.1 – Example for the extraction of gridness and orientation.** In a first step, the weight map (left) of a cell is taken and the spatial auto-correlation computed (middle). Afterwards, a ring with the expected hexagonal peaks is extracted (right). The ring is correlated with rotated versions of itself to extract the gridness according to Equation (B.2). Furthermore, the orientation of the cell is computed with respect to the horizontal axis.

The entire procedure is depicted in Figure B.1. It shows a weight map for a cell on the left. In the middle, the auto-correlogram is depicted with the area which is extracted. The extracted ring, shown on the right, is correlated with rotated versions of itself to compute the gridness. Furthermore, the orientation of the cell is computed based on the first peak above zero in counter-clockwise direction as visualized.

## Appendix C

# Preliminary results in favor of the dense sampling assumption

It is certainly arguable if there is a sensory representation which allows optimal hexagonal packing of grid fields. To examine this issue, a simulation was set up in which an artificial neuron samples from the boundary vector space. Note that the model and data reported in this section are only preliminary and will require thorough analysis in a future study.

Taking inspiration from the results of Chapter 6, in particular Section 6.1 and Section 6.2, each receptive field of the artificial neuron was modelled in form of a particle  $p_i$  with repulsive and attractive fields. Thereby the particles expose dynamics similar to on-center/off-surround receptive fields as suggested in Section 6.1. Furthermore, the entire aggregate of particles can be understood as a dendritic tree computation performed within a single neuron such as presented in Section 6.2.

The input to the neuron is a normalized boundary vector of dimension  $N_{bv}$ . For instance,  $N_{bv} = 10$  means that the agent has access to 10 boundary distance estimates encoded in a vector of dimension 10. Subsequently, the vector is normalized such that its total length is 1.

The level of activation  $a_i$  of particle  $p_i$  depends on the distance between its receptive vector  $c_i$  and the boundary vector  $v_x$  at location  $x \in \mathbb{R}^2$ , i.e.

$$a_i = 1 - \sqrt{\sum_j^{N_{bv}} (c_{i,j} - v_{x,j})^2}, \quad (\text{C.1})$$

where subscript  $j$  indicates the  $j$ -th entry in a vector. The winning particle is then simply the one which maximizes  $a_i$ , i.e.  $a_w = \max_i (a_i)$  and  $w = \arg \max_i (a_i)$ . Additionally, all particles above a certain threshold  $a_{\text{thresh}}$  are collected, i.e.  $P_{\text{win}} = \{i | a_i \geq a_{\text{thresh}}\}$ .

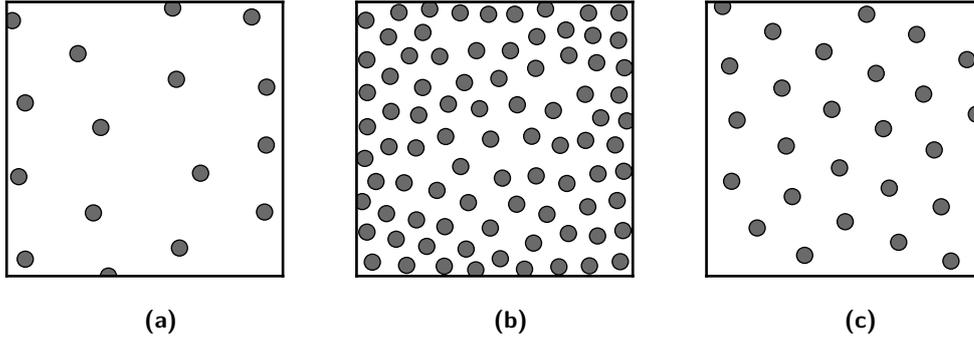
If the activation  $a_w < a_{\text{thresh}}$ , then a novel particle is introduced and assigned to  $v_x$ . On the other hand, if  $a_w \geq a_{\text{thresh}}$ , then  $c_w$  is updated according to  $c_w \leftarrow c_w + \alpha(v_x - c_w)$  with the plasticity factor or learning rate  $\alpha$ . All other particles  $i \in P_{\text{win}}, i \neq w$  are pushed away from  $v_x$  according to  $c_i \leftarrow c_i + \beta(c_i - v_x)$ .

In all simulations, the experimental environment was a rectangular box with walls of length 1.0 m. The input statistics of the virtual agent, i.e. the distribution of velocities and turning directions, was kept close to data from real rats. The activation of any particle was tracked over time with respect to its two dimensional coordinate.

### C. Preliminary results in favor of the dense sampling assumption

**Table C.1 – Parameters for sampling boundary vectors.** The sampling process uses particles with attractive and repulsive fields.

	$N_{bv}$	$a_{thresh}$	$\alpha$	$\beta$
Figure C.1a	8	0.200	0.001	0.001
Figure C.1b	8	0.200	0.002	0.010



**Figure C.1 – Preliminary results of sampling from the boundary vector space in an arena of size  $1 \times 1 \text{m}^2$ .** Each circle denotes the center of the area in which a particle  $p_i$  is active due to its assigned boundary vector  $c_i$ . The distribution of sampling centers is not perfectly hexagonal yet. **(a)** and **(b)** differ in the size of the receptive fields and thus the strength of attraction and repulsion. **(c)** is a result of the particle sampling process applied to two dimensional coordinates.

The results suggest that sampling from the boundary vector space is one likely element to generate hexagonal grid cell fields. Examples for the parameters listed in Table C.1 are depicted in Figure C.1a and b. The results are not yet perfectly hexagonal everywhere, however this may be due to unsuitable parameters or because the model is too simple at the moment. Another reason could be the learning parameters  $\alpha$  and  $\beta$  which are kept constant throughout the simulation.

In simulations with smaller receptive fields and therefore more particles (or a smaller grid cell period), hexagonality is predominantly preserved within quadrants of the whole area and fractures in between (see Figure C.1b for an example). Further visual inspection of the data indicates that effects which were observed in real grid cells may be present, namely a slight angular rotation with respect to boundaries as well as skewing and shearing [198, 337]. The effects were attributed to increase the asymmetry within the grid representation, however it is unclear what the purpose of the asymmetry would be. They could however also be the result of a dynamic process densely packing transitions in the sensor space which in turn leads to hexagons in the typically used symmetric experimental environments. To test this assumption, the same particles as described above were used to sample from two dimensional coordinates instead of the boundary vector space. The regularity of coordinates and independence of a sensory state should generate perfect hexagons in the whole arena. Due to the receptive field dynamics, the process should also yield a close to optimal filling of the area with receptive field centers. The results, visualized in Figure C.1c, indicate that this could indeed be the origin of the orientation offset.

The important next step is to investigate the dynamics required to generate proper

hexagonal fields and to analyse the already available data in depth. Furthermore, it is necessary to determine the real driving inputs to the grid cell system. A complex model of the implied dendritic computation of a grid cell with feed-forward input from boundary cells is currently under development and will presumably lead to novel insights. As was discussed already in Section 6.3, the boundary vector space in combination with learning that also depends on a speed signal, such as expressed by speed cells [195], likely stabilizes the formation of the fields. Finally, the simulations will be extended to non-regular environments.



## Appendix D

# Algorithms for Sparse Layered Graphs and Transition Graphs

The algorithms in this section are reprinted with permission from [22, 372].

---

### Algorithm 1 Generation of the Sparse Layered Graph

---

```

Initialize:  $i = 1, \mathbf{S}_0 = \mathcal{G}$ 
for all  $v \in V_0^{\mathbf{S}}$  do ▷ clustering on layer 0
    Assign each  $v$  to closest cell  $c^j$ 
end for
while  $i \leq L - 1$  do ▷ construction of layers  $i > 0$ 
     $\mathbf{S}_i = \mathbf{S}_{i-1}$ 
    for all  $v \in V_i^{\mathbf{S}}$  do ▷ clustering on layer  $i$ 
        Assign each  $v$  to closest cell  $c^j$ 
    end for
    for all  $c^j, j \in J$  do ▷ node contraction in each cell
        for all contractable nodes  $v$  with  $\deg(v) \geq 2$  do
            for all neighbor pairs  $(u, w)$  of  $v$  do
                 $d^{contracted} \leftarrow d(u, v) + d(v, w)$ 
                if  $(u, w) \in E_i^{\mathbf{S}} \wedge d^{contracted} < d(u, w)$  then
                     $d(u, w) \leftarrow d^{contracted}$ 
                    flag edge  $(u, w)$  as contracted
                else if  $(u, w) \notin E_i^{\mathbf{S}}$  then
                    add new edge  $(u, w)$  to  $\mathbf{S}_i$ 
                     $d(u, w) \leftarrow d^{contracted}$ 
                    flag edge  $(u, w)$  as contracted
                end if
            end for
            remove  $v$  from  $\mathbf{S}_i$ 
            mark all neighbors of  $v$  as non-contractable
        end for
    end for
    end for
     $i \leftarrow i + 1$ 
end while

```

---

**Algorithm 2** Generation of the Transition Graph

---

```

for all  $i \in I$  do
  Initialize:  $\mathbf{T}_i \leftarrow (B_i, F_i)$  ▷ see Definition 8
  for all cells  $c^j, j \in J$  do ▷ all-pair shortest-path
    Initialize:  $\mathbf{T}_i^j \leftarrow (\emptyset, \emptyset)$ 
    for all  $u, v \in B_i^j$  do
       $w(u, v) \leftarrow \min d(u, w)$  in  $\mathbf{S}_i^j$ 
       $\mathbf{T}_i^j = \mathbf{T}_i^j \cup e(u, v)$  ▷ insert (new) edge
      flag edge  $e(u, v)$  as an APSP-edge
    end for
     $\mathbf{T}_i = \mathbf{T}_i \cup \mathbf{T}_i^j$ 
  end for
end for

```

---

**Algorithm 3** Shortest distance query: `getDist`


---

```

Input source and target nodes  $s, t \in V$ 
Output  $dist$  - length of shortest path between  $s$  and  $t$ 
 $k \leftarrow$  lowest common level of  $s$  and  $t$ 
 $[c_s, c_t] \leftarrow$  cells on level  $k$  which include  $s$  and  $t$ 
 $\mathbf{Q} = \mathbf{T}_k \cup \mathbf{S}_k^{c_s} \cup \mathbf{S}_k^{c_t}$  ▷ merge sub-graphs
 $P \leftarrow$  shortest path in  $\mathbf{Q}$  ▷ for instance with Dijkstra return  $|P|$ 

```

---

**Algorithm 4** Shortest path query: `getPath`


---

```

Input source and target nodes  $s, t \in V$ 
Output  $P^0$  - shortest path between  $s$  and  $t$ 
 $k \leftarrow$  lowest common level of  $s$  and  $t$ 
 $[c_s, c_t] \leftarrow$  cells on level  $k$  which include  $s$  and  $t$ 
 $\mathbf{Q} = \mathbf{T}_k \cup \mathbf{S}_k^{c_s} \cup \mathbf{S}_k^{c_t}$  ▷ merge sub-graphs
 $P_Q \leftarrow$  shortest path in  $\mathbf{Q}$  ▷ for instance with Dijkstra
 $P^k \leftarrow P_Q$  with expanded APSP edges ▷ 1. phase
 $P^0 \leftarrow P^k$  with recursively expanded edges ▷ 2. phase
return  $P^0$ 

```

---

## Appendix E

# On the complexity of algorithms for SLGs and TGs

The following derivations are reprinted with permission from [22, 372].

### E.1 Construction algorithms

Two tables are required in a naive implementation of the construction of SLG, one which contains edge and the other for node information. After sorting each of the tables, binary search yields logarithmic access times for arbitrary elements. The pre-processing time complexity is thus given by

$$\mathcal{O}(E \log E + V \log V). \quad (\text{E.1})$$

In non-degenerate cases of the input graph, the number of vertices is assumed to be lower than the number of edges. For instance, many real-world graphs can be considered scale free, i.e. a single node of a graph is typically connected to only few neighbors. Thus, Equation E.1 can be simplified to

$$\begin{aligned} \mathcal{O}(E \log E + V \log V) &\subseteq \mathcal{O}(2 \cdot (E \log E)) \\ &\subseteq \mathcal{O}(E \log E). \end{aligned} \quad (\text{E.2})$$

Assuming a scale-free network allows to implement the algorithm with plain arrays to store nodes, and adjacency lists for edge information. Sorting such an array requires  $\mathcal{O}(V \log V)$  time, whereas any adjacency list can be sorted in at most  $\mathcal{O}(\deg_m(V) \log(\deg_m(V)))$  time. Here  $\deg_m(V) := \max\{\deg(V)\}$ . An amortized run-time complexity to sort the data can be stated based on the assumption that  $1 \leq \deg(v) \ll V$  in most real-world graphs, i.e.

$$\begin{aligned} &\mathcal{O}(V \log V + V \cdot \deg_m(V) \cdot \log(\deg_m(V))) \\ \stackrel{\text{amortized}}{\subseteq} &\mathcal{O}(V \log V + V \log V) \subseteq \mathcal{O}(2 \cdot V \log V) \end{aligned} \quad (\text{E.3})$$

Further, the assumption allows to assess the amortized retrieval time as

$$\mathcal{O}(1 + \log V) \subseteq \mathcal{O}(\log V) \quad (\text{E.4})$$

for adjacency information for any node.

## E. On the complexity of algorithms for SLGs and TGs

To construct the next layer, node contraction is performed. Consider a single node  $v \in V$ , and any pair of neighbors  $u, w \in V$  of  $v$ . Contraction requires the removal of any edge from  $u$  and  $w$  to  $v$  which requires at most  $\mathcal{O}(\deg(v))$  operations. Combined with E.4 this yields

$$\mathcal{O}(\log V + \deg(v)) \subseteq \mathcal{O}(\log V + E) \quad (\text{E.5})$$

in the worst case to contract a single node  $v$ . In the case of a scale-free network, i.e.  $1 \leq \deg(V) \ll V \leq E$ , the run-time is bounded from below by

$$\mathcal{O}(\log V + \deg(v)) \stackrel{\text{amortized}}{\supseteq} \mathcal{O}(\log V + 1). \quad (\text{E.6})$$

To assess the worst case, two scenarios have to be considered. First assume that  $u, w \in V$  are the only neighbors of  $v$ . Then, the lower bound of the time complexity is governed by Equation E.5. As all neighbors  $u, w$  of  $v$  are marked as *non-contractable* during contraction, this operation is performed at most  $\frac{V}{2}$  times. This yields the overall lower bound to contract all nodes  $v \in V$  and within a single cell of

$$\mathcal{O}\left(\frac{V}{2}E\right) \subseteq \mathcal{O}(V \cdot E) \stackrel{V \leq E}{\subseteq} \mathcal{O}(E^2). \quad (\text{E.7})$$

For the amortized case motivated in Equation E.6 the bound is given by

$$\mathcal{O}(V \cdot (\log V + 1)) \stackrel{\text{amortized}}{\subseteq} \mathcal{O}(V \cdot \log V + V). \quad (\text{E.8})$$

Now consider the case in which  $v$  is connected to all other nodes  $u, w \in V$ , but  $\nexists e(u, w) \in E$  for any  $u, w$ . Thus, the graph has a star topology and  $\deg(v) = E$ . Then, contraction of  $v$  requires to introduce  $\frac{E(E-1)}{2}$  novel and removal of  $E$  old edges. Consequently, the upper bound estimate is governed by a cubic term, i.e.

$$\mathcal{O}\left(E + E \frac{E(E-1)}{2}\right) \subseteq \mathcal{O}(E^3). \quad (\text{E.9})$$

A suitable data structure, e.g. a hash table, can improve the estimate to

$$\mathcal{O}\left(E + \frac{E(E-1)}{2}\right) \subseteq \mathcal{O}(E^2). \quad (\text{E.10})$$

As any other node  $u \in V$  is subsequently marked as non-contractable, this also states a tight upper bound for the amortized case.

The analysis is now extended to multiple scales by considering the *naive* worst case run-time complexity  $\Omega$  and the *amortized* worst-case complexity  $\Omega_{\text{amortized}}$ . Equations E.7 and E.10 show that the latter is approached by  $\mathcal{O}(E^2)$  both from bottom and top. Hence the focus can be restricted to the additional steps required to introduce a novel layer.

Sequentially sorting all existing cells requires  $\mathcal{O}(N_i \log N_i)$  time for  $N_i$  cells on layer  $i$ , and assigning any  $v \in V_i$  to the closest cluster can be achieved in  $\mathcal{O}(V_i \log N_i)$  time with the help of a suitable data structure, e.g. a kd-tree. It is also assumed that  $L$  is a small number, i.e.  $N_i \ll V_i$  by which  $N_i \ll E_i$ , and that  $N_i \leq N_{i+1}$  holds. In the following,  $E := \max_i E_i$  and  $V := \max_i V_i$  will be used. The worst-case run-time

complexity is thus the combination of sorting the cells on the lowest layer on which no contraction is performed, and subsequently contracting nodes on all subsequent layers. Hence it is given by

$$\begin{aligned} \Omega &= \mathcal{O}(N_0 \log N_0 + V_0 \log N_0 + \\ &\quad \sum_{i=1}^{L-1} (N_i \log N_i + V_i \log N_i + N_i \cdot E_i^2)) \\ &\subseteq \mathcal{O}(L \cdot E^2) \subseteq \mathcal{O}(E^2). \end{aligned} \quad (\text{E.11})$$

The run-time complexity is thus governed by the maximal number of edges  $E$  in  $\mathbf{S}$ , as  $N_i$  and  $L$  become negligible for a large  $E$ . The analysis for the amortized case yields a run-time complexity of

$$\mathcal{O}(V \log V + E) \subseteq \Omega_{\text{amortized}} \subseteq \mathcal{O}(E^2). \quad (\text{E.12})$$

The analysis of the construction of TG  $\mathbf{T}$  follows likewise. First, the analysis will be stated for an arbitrary cell  $c$  on layer  $i$ , for which the node and edge sets are denoted as  $V_i^c$  and  $E_i^c$ , respectively.

Using a lookup table,  $\mathbf{S}_i^c$  can be extracted in  $O(1)$  time. Similarly, the identification of any border node requires  $O(1)$  time. Dijkstra's algorithm is used to compute the all-pair-shortest path, thus

$$B_i^c(E_i^c + V_i^c \log V_i^c) \quad (\text{E.13})$$

is the time complexity for the construction of  $c$ .

Using the assumption of a scale-free network as above leads to the approximate values

$$V_i^c \approx \frac{V_i^{\mathbf{S}}}{C_i}, \quad E_i^c \approx \frac{E_i^{\mathbf{S}}}{C_i}, \quad B_i^c \leq B_i \quad (\text{E.14})$$

for any cell  $c$ . The overall run-time complexity of the construction of  $\mathbf{T}$  follows accordingly as

$$\begin{aligned} &\mathcal{O}\left(\sum_{i=0}^{L-1} C_i B_i \left(\frac{E_i^{\mathbf{S}}}{C_i} + \frac{V_i^{\mathbf{S}}}{C_i} \log \frac{V_i^{\mathbf{S}}}{C_i}\right)\right) \\ &= \mathcal{O}\left(\sum_{i=0}^{L-1} B_i \left(E_i^{\mathbf{S}} + V_i^{\mathbf{S}} \log \frac{V_i^{\mathbf{S}}}{C_i}\right)\right) \end{aligned} \quad (\text{E.15})$$

For the amortized case in which  $B_i \ll V_i$ , this reduces to

$$\begin{aligned} &\mathcal{O}\left(\sum_{i=0}^{L-1} E_i^{\mathbf{S}} + V_i^{\mathbf{S}} \log \frac{V_i^{\mathbf{S}}}{C_i}\right) \\ &\subseteq \mathcal{O}\left(\sum_{i=0}^{L-1} E_i^{\mathbf{S}} + V_i^{\mathbf{S}} \log V_i^{\mathbf{S}}\right) \\ &\subseteq \mathcal{O}(L \cdot (E + V \log V)), \end{aligned} \quad (\text{E.16})$$

where  $E := \max E_i^{\mathbf{S}}$  and  $V := \max V_i^{\mathbf{S}}$

## E.2 Retrieval algorithms

In the worst case for shortest distance queries, the nodes are contracted already on the smallest scale. Then query will be performed on the entire graph. However, in all other cases the computation of the shortest path using Dijkstra's algorithm on a TG is at least as fast as applying the algorithm to the SLG. This is due to the construction of the TG which ideally sparsifies SLG. This depends on the implementation of the union operator, though. The operator is required to merge cells of the SLG and TG as mentioned in the description of the algorithm.

Retrieval of a shortest path requires the expansion of contracted as well as all-pair-shortest-path edges. Denote the aforementioned union of cells of the SLG  $\mathbf{S}$  with the TG  $\mathbf{T}$  by the temporary graph  $\mathbf{Q}$ , and the shortest path in  $\mathbf{Q}$  as  $P_{\mathbf{Q}}$ . Expanding any all-pair-shortest-path on layer  $k$  yields the shortest path  $P_{\mathbf{S}}^k$  in  $\mathbf{S}_k$ . Furthermore,  $P_{\mathbf{Q}}$  crosses  $\zeta^{\mathbf{Q}}$  many cells in  $\mathbf{Q}$ , with  $0 \leq \zeta^{\mathbf{Q}} \leq C_k - 1$  and  $C_k$  the number of cells on layer  $k$ . Then, the run-time complexity of shortest path retrieval is given by

$$\mathcal{O}(\zeta^{\mathbf{Q}} \left( \frac{E_k^{\mathbf{S}}}{C_k} + \frac{V_k^{\mathbf{S}}}{C_k} \log \frac{V_k^{\mathbf{S}}}{C_k} \right)). \quad (\text{E.17})$$

Simulations suggest that that  $\zeta^{\mathbf{Q}} \approx \frac{C_k}{2}$ . In this case, Equation E.17 can be assumed to follow according to

$$\mathcal{O}(E_k^{\mathbf{S}} + V_k^{\mathbf{S}} \log \frac{V_k^{\mathbf{S}}}{C_k}) \subseteq \mathcal{O}(E_k^{\mathbf{S}} + V_k^{\mathbf{S}} \log V_k^{\mathbf{S}}). \quad (\text{E.18})$$

Recursively expanding all remaining contracted edges requires at most  $2^{k-1} \cdot (P_{\mathbf{S}}^k - 1)$  computations in lower-level cells.

## Bibliography

- [1] R. Agrawal and H. V. Jagadish. Efficient search in very large databases. In *Proceedings of the 14th International Conference on Very Large Data Bases, VLDB '88*, pages 407–418, San Francisco, CA, USA, 1988. Morgan Kaufmann Publishers Inc.
- [2] R. Aldecoa and I. Marin. Deciphering network community structure by surprise. *PLoS ONE*, 6(9):e24195, 2011.
- [3] R. Aldecoa and I. Marin. Surprise maximization reveals the community structure of complex networks. *Sci Rep*, 3:1060, 2013.
- [4] S. Amari. Dynamics of pattern formation in lateral-inhibition type neural fields. *Biol Cybern*, 27(2):77–87, 8 1977.
- [5] D. J. Amit. *Modeling Brain Function – the World of Attractor Neural Networks*. Cambridge University Press, New York, NY, USA, 1989.
- [6] P. Andersen. Organization of hippocampal neurons and their interconnections. In R. L. Isaacson and K. H. Pribram, editors, *The Hippocampus: Volume 1: Structure and Development*, pages 155–175. Springer US, Boston, MA, 1975.
- [7] P. Andersen, R. Morris, and D. Amaral, editors. *The Hippocampus Book (Oxford Neuroscience Series)*. Oxford University Press, 2006.
- [8] C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. Pyramid methods in image processing, 1984.
- [9] C. Armstrong, J. Wang, S. Yeun Lee, J. Broderick, M. J. Bezaire, S. H. Lee, and I. Soltesz. Target-selectivity of parvalbumin-positive interneurons in layer II of medial entorhinal cortex in normal and epileptic animals. *Hippocampus*, 26(6):779–793, Jun 2016.
- [10] G. A. Ascoli, S. Gasparini, V. Medinilla, and M. Migliore. Local control of postinhibitory rebound spiking in ca1 pyramidal neuron dendrites. *J Neurosci*, 30(18):6434–6442, May 2010.
- [11] J. S. Bains, J. M. Longacher, and K. J. Staley. Reciprocal interactions between ca3 network activity and strength of recurrent collateral synapses. *Nat Neurosci*, 2(8):720–726, Aug 1999.

- [12] C. A. Barnes. Spatial learning and memory processes: the search for their neurobiological mechanisms in the rat. *Trends Neurosci.*, 11(4):163–169, Apr 1988.
- [13] C. Barry, C. Lever, R. Hayman, T. Hartley, S. Burton, J. O’Keefe, K. Jeffery, and N. Burgess. The boundary vector cell model of place cell firing and spatial memory. *Rev Neurosci*, 17(1-2):71–97, 2006.
- [14] C. Barry, L. L. Ginzberg, J. O’Keefe, and N. Burgess. Grid cell firing patterns signal environmental novelty by expansion. *Proceedings of the National Academy of Sciences*, 109(43):17687–17692, 2012.
- [15] C. Barry, R. Hayman, N. Burgess, and K. J. Jeffery. Experience-dependent rescaling of entorhinal grids. *Nat Neurosci*, 10(6):682–684, Jun 2007.
- [16] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck. Route planning in transportation networks, 2015.
- [17] H. Bast, S. Funke, D. Matijevic, P. Sanders, and D. Schultes. In transit to constant time shortest-path queries in road networks. In *Proceedings of the Meeting on Algorithm Engineering & Experiments*, pages 46–59, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.
- [18] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-Up Robust Features (SURF). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008.
- [19] S. Behnke and R. Rojas. Neural abstraction pyramid: a hierarchical image understanding architecture. In *1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence*, volume 2, pages 820–825 vol.2, May 1998.
- [20] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, September 1975.
- [21] P. Berkes and L. Wiskott. Slow feature analysis yields a rich repertoire of complex cell properties. *J Vis*, 5(6):579–602, Jul 2005.
- [22] E. Berzs. Distributed shortest path calculation with a hybrid data structure. Master’s thesis, Technische Universität München, 2016.
- [23] G. Bi and M. Poo. Synaptic modification by correlated activity: Hebb’s postulate revisited. *Annu. Rev. Neurosci.*, 24:139–166, 2001.
- [24] C. Bielza and P. Larrañaga. Bayesian networks in neuroscience: a survey. *Frontiers in Computational Neuroscience*, 8:131, 2014.
- [25] E. L. Bienenstock, L. N. Cooper, and P. W. Munro. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J. Neurosci.*, 2(1):32–48, Jan 1982.

- [26] D. K. Bilkey. The Parahippocampal Region: Organization and role in cognitive function. *Brain*, 127(5):1211–1212, feb 2004.
- [27] T. L. Bjercknes, E. I. Moser, and M. B. Moser. Representation of geometric borders in the developing rat. *Neuron*, 82(1):71–78, Apr 2014.
- [28] T. Bonnevie, B. Dunn, M. Fyhn, T. Hafting, D. Derdikman, J. L. Kubie, Y. Roudi, E. I. Moser, and M.-B. Moser. Grid cells require excitatory drive from the hippocampus. *Nat Neurosci*, 16(3):309–317, Mar 2013.
- [29] M. Borst, A. Knoblauch, and G. Palm. Modelling the auditory system: pre-processing and associative memories using spiking neurons. *Neurocomputing*, 58-60:1013–1018, jun 2004.
- [30] M. Bosch and Y. Hayashi. Structural plasticity of dendritic spines. *Curr. Opin. Neurobiol.*, 22(3):383–388, Jun 2012.
- [31] C. Boucheny, N. Brunel, and A. Arleo. A continuous attractor network model without recurrent excitation: maintenance and integration in the head direction cell system. *J Comput Neurosci*, 18(2):205–227, 2005.
- [32] R. Brette and W. Gerstner. Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *J. Neurophysiol.*, 94(5):3637–3642, Nov 2005.
- [33] K. Brodmann. Vergleichende Lokalisationslehre der Grosshirnrinde in ihren Prinzipien dargestellt auf Grund des Zellenbaues. *Leipzig: Johann Ambrosius Barth*, 1909.
- [34] S. P. Brown, S. He, and R. H. Masland. Receptive field microstructure and dendritic geometry of retinal ganglion cells. *Neuron*, 27(2):371 – 383, 2000.
- [35] D. Brüderle, M. A. Petrovici, B. Vogginger, M. Ehrlich, T. Pfeil, S. Millner, A. Grübl, K. Wendt, E. Müller, M.-O. Schwartz, D. H. de Oliveira, S. Jeltsch, J. Fieres, M. Schilling, P. Müller, O. Breitwieser, V. Petkov, L. Muller, A. P. Davison, P. Krishnamurthy, J. Kremkow, M. Lundqvist, E. Muller, J. Partzsch, S. Scholze, L. Zühl, C. Mayr, A. Destexhe, M. Diesmann, T. C. Potjans, A. Lansner, R. Schüffny, J. Schemmel, and K. Meier. A comprehensive workflow for general-purpose neural modeling with highly configurable neuromorphic hardware systems. *Biological Cybernetics*, 104(4):263–296, 2011.
- [36] C. Buetfering, K. Allen, and H. Monyer. Parvalbumin interneurons provide grid cell-driven recurrent inhibition in the medial entorhinal cortex. *Nat Neurosci*, 17(5):710–718, May 2014. Article.
- [37] E. Bullmore and O. Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nat Rev Neurosci*, 10(3):186–198, Mar 2009.

- [38] A. Buluç, H. Meyerhenke, I. Safro, P. Sanders, and C. Schulz. Recent advances in graph partitioning. In L. Kliemann and P. Sanders, editors, *Algorithm Engineering: Selected Results and Surveys*, pages 117–158. Springer International Publishing, Cham, 2016.
- [39] Y. Burak and I. R. Fiete. Accurate path integration in continuous attractor network models of grid cells. *PLoS Computational Biology*, 5(2):1–16, 02 2009.
- [40] N. Burgess, C. Barry, and J. O’Keefe. An oscillatory interference model of grid cell firing. *Hippocampus*, 17(9):801–812, 2007.
- [41] P. Burt and E. Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4):532–540, Apr 1983.
- [42] D. Bush, C. Barry, and N. Burgess. What do grid cells contribute to place cell firing? *Trends Neurosci.*, 37(3):136–145, Mar 2014.
- [43] D. A. Butts and M. S. Goldman. Tuning curves, neuronal variability, and sensory coding. *PLoS Biol.*, 4(4):e92, Apr 2006.
- [44] G. Buzsáki. Two-stage model of memory trace formation: a role for ”noisy” brain states. *Neuroscience*, 31(3):551–570, 1989.
- [45] G. Buzsáki. Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning. *Hippocampus*, 25(10):1073–1188, Oct 2015.
- [46] G. Buzsáki. Neural syntax: cell assemblies, synapsembles and readers. *Neuron*, 68(3):362–385, Nov 2010.
- [47] G. Buzsáki and E. I. Moser. Memory, navigation and theta rhythm in the hippocampal-entorhinal system. *Nat Neurosci*, 16(2):130–138, Feb 2013.
- [48] G. Buzsáki. Theta oscillations in the hippocampus. *Neuron*, 33(3):325 – 340, 2002.
- [49] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, Dec 2016.
- [50] C. B. Canto, F. G. Wouterlood, and M. P. Witter. What does the anatomical organization of the entorhinal cortex tell us? *Neural Plasticity*, 2008:1–18, 2008.
- [51] G. A. Carpenter and S. Grossberg. The art of adaptive pattern recognition by a self-organizing neural network. *Computer*, 21(3):77–88, March 1988.
- [52] M. A. Castro-Alamancos and B. W. Connors. Distinct forms of short-term plasticity at excitatory synapses of hippocampus and neocortex. *Proceedings of the National Academy of Sciences*, 94:4161, 1997.
- [53] S. Chamberland and L. Topolnik. Inhibitory control of hippocampal inhibitory neurons. *Frontiers in Neuroscience*, 6:165, 2012.

- [54] R. Chaudhuri and I. Fiete. Computational principles of memory. *Nat Neurosci*, 19(3):394–403, Mar 2016. Review.
- [55] G. Chen, D. Manson, F. Cacucci, and T. J. Wills. Absence of visual input results in the disruption of grid cell firing in the mouse. *Current Biology*, 26(17):2335 – 2342, 2016.
- [56] L. L. Chen, L.-H. Lin, E. J. Green, C. A. Barnes, and B. L. McNaughton. Head-direction cells in the rat posterior cortex. *Experimental Brain Research*, 101(1):8–23, 1994.
- [57] T.-W. Chen, T. J. Wardill, Y. Sun, S. R. Pulver, S. L. Renninger, A. Baohan, E. R. Schreiter, R. A. Kerr, M. B. Orger, V. Jayaraman, L. L. Looger, K. Svoboda, and D. S. Kim. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, 499(7458):295–300, Jul 2013. Article.
- [58] S. Cheng. The crisp theory of hippocampal function in episodic memory. *Frontiers in Neural Circuits*, 7:88, 2013.
- [59] F. Chersi and N. Burgess. The Cognitive Architecture of Spatial Navigation: Hippocampal and Striatal Contributions. *Neuron*, 88(1):64–77, Oct 2015.
- [60] A. Chin. Locality-preserving hash functions for general purpose parallel computation. *Algorithmica*, 12(2):170–181, 1994.
- [61] H. Choset and K. Nagatani. Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization. *IEEE Transactions on Robotics and Automation*, 17(2):125–137, Apr 2001.
- [62] P. S. Churchland, C. Koch, and T. J. Sejnowski. What is computational neuroscience? In E. L. Schwartz, editor, *Computational Neuroscience*, pages 46–55. MIT Press, Cambridge, MA, USA, 1993.
- [63] J. Cichon and W.-B. Gan. Branch-specific dendritic Ca<sup>2+</sup> spikes cause persistent synaptic plasticity. *Nature*, 520(7546):180–185, Apr 2015. Article.
- [64] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber. Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32:333 – 338, 2012. Selected Papers from {IJCNN} 2011.
- [65] H. Cohn. A conceptual breakthrough in sphere packing. *Notices of the American Mathematical Society*, 64(02):102–115, feb 2017.
- [66] H. Cohn, A. Kumar, S. D. Miller, D. Radchenko, and M. Viazovska. The sphere packing problem in dimension 24, 2016.
- [67] L. L. Colgin, E. I. Moser, and M. B. Moser. Understanding memory through hippocampal remapping. *Trends Neurosci.*, 31(9):469–477, Sep 2008.
- [68] J. H. Conway, N. J. A. Sloane, and E. Bannai. *Sphere-packings, Lattices, and Groups*. Springer-Verlag New York, Inc., New York, NY, USA, 1987.

- [69] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009.
- [70] J. J. Couey, A. Witoelar, S. J. Zhang, K. Zheng, J. Ye, B. Dunn, R. Czajkowski, M. B. Moser, E. I. Moser, Y. Roudi, and M. P. Witter. Recurrent inhibitory circuitry as a mechanism for grid formation. *Nat. Neurosci.*, 16(3):318–324, Mar 2013.
- [71] M. Cross. *Pattern formation and dynamics in nonequilibrium systems*. Cambridge University Press, Cambridge, UK New York, 2009.
- [72] J. Csicsvari and D. Dupret. Sharp wave/ripple network oscillations and learning-associated hippocampal maps. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 369(1635):20120528, Feb 2014.
- [73] N. Cuperlier, P. Laroque, P. Gaussier, and M. Quoy. Planning and navigation strategies using transition cells and neural fields. In *Proc. of ASC/IASTED 2004 (Artificial intelligence and Soft Computing / International Association of Science and TEchnology for Development)*, Marbella, sept 2004.
- [74] N. Cuperlier, M. Quoy, C. Giovannangeli, P. Gaussier, and P. Laroque. Transition cells for navigation and planning in an unknown environment. In S. Nolfi, G. Baldassarre, R. Calabretta, J. C. T. Hallam, D. Marocco, J.-A. Meyer, O. Miglino, and D. Parisi, editors, *From Animals to Animats 9: 9th International Conference on Simulation of Adaptive Behavior, SAB 2006, Rome, Italy, September 25-29, 2006. Proceedings*, pages 286–297, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [75] Y. Dabaghian, F. Memoli, L. Frank, and G. Carlsson. A topological paradigm for hippocampal spatial map formation using persistent homology. *PLoS Comput. Biol.*, 8(8):e1002581, 2012.
- [76] Y. Dabaghian, V. L. Brandt, and L. M. Frank. Reconceiving the hippocampal map as a topological template. *eLife*, 3:e03476, aug 2014.
- [77] V. L. DalleMole and A. F. R. Araújo. A novel topological map of place cells for autonomous robots. In K. Diamantaras, W. Duch, and L. S. Iliadis, editors, *Artificial Neural Networks – ICANN 2010: 20th International Conference, Thessaloniki, Greece, September 15-18, 2010, Proceedings, Part II*, pages 296–306, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [78] Y. Dan and M. M. Poo. Spike timing-dependent plasticity of neural circuits. *Neuron*, 44(1):23–30, Sep 2004.
- [79] P. Dayan and N. D. Daw. Decision theory, reinforcement learning, and the brain. *Cogn Affect Behav Neurosci*, 8(4):429–453, Dec 2008.
- [80] P. Dayan and L. F. Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. The MIT Press, 2001.

- [81] D. Debanne, B. H. Gähwiler, and S. M. Thompson. Asynchronous pre- and postsynaptic activity induces associative long-term depression in area CA1 of the rat hippocampus in vitro. *Proc. Natl. Acad. Sci. U.S.A.*, 91(3):1148–1152, Feb 1994.
- [82] D. Delling, M. Holzer, K. Müller, F. Schulz, and D. Wagner. High-performance multi-level graphs. In *In: 9th DIMACS Implementation Challenge*, pages 52–65, 2006.
- [83] D. Delling, M. Holzer, K. Müller, F. Schulz, and D. Wagner. High-performance multi-level routing, 2008.
- [84] D. Derdikman and E. I. Moser. A manifold of spatial maps in the brain. *Trends Cogn. Sci. (Regul. Ed.)*, 14(12):561–569, Dec 2010.
- [85] D. Derdikman, J. R. Whitlock, A. Tsao, M. Fyhn, T. Hafting, M.-B. Moser, and E. I. Moser. Fragmentation of grid cell maps in a multicompartment environment. *Nat Neurosci*, 12(10):1325–1332, Oct 2009.
- [86] K. Diba, A. Amarasingham, K. Mizuseki, and G. Buzsáki. Millisecond timescale synchrony among hippocampal neurons. *Journal of Neuroscience*, 34(45):14984–14994, 2014.
- [87] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, dec 1959.
- [88] K. Doya, S. Ishii, A. Pouget, and R. P. Rao, editors. *Bayesian Brain – Probabilistic Approaches to Neural Coding*. MIT Press - Journals, dec 2006.
- [89] D. Dupret, B. Pleydell-Bouverie, and J. Csicsvari. Rate remapping: When the code goes beyond space. *Neuron*, 68(6):1015 – 1016, 2010.
- [90] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics Automation Magazine*, 13(2):99–110, June 2006.
- [91] J. C. Eccles. Interpretation of action potentials evoked in the cerebral cortex. *Electroencephalogr Clin Neurophysiol*, 3(4):449–464, Nov 1951.
- [92] R. Eckhorn, R. Bauer, W. Jordan, M. Brosch, W. Kruse, M. Munk, and H. J. Reitboeck. Coherent oscillations: A mechanism of feature linking in the visual cortex? *Biological Cybernetics*, 60(2):121–130, 1988.
- [93] V. Edvardsen. A passive mechanism for goal-directed navigation using grid cells. In *Proceedings of the European Conference on Artificial Life 2015*. The MIT Press, jul 2015.
- [94] M. J. Egenhofer. What’s special about spatial? database requirements for vehicle navigation in geographic space (extended abstract). In P. Buneman and S. Jajodia, editors, *SIGMOD Conference*, pages 398–402. ACM Press, 1993.
- [95] G. Ehret. The auditory cortex. *Journal of Comparative Physiology A*, 181(6):547–557, 1997.

- [96] H. Eichenbaum. Time cells in the hippocampus: a new dimension for mapping memories. *Nat. Rev. Neurosci.*, 15(11):732–744, Nov 2014.
- [97] H. Eichenbaum, P. Dudchenko, E. Wood, M. Shapiro, and H. Tanila. The hippocampus, memory, and place cells: Is it spatial memory or a memory space? *Neuron*, 23(2):209 – 226, 1999.
- [98] S. L. Epstein. Wanted: Collaborative intelligence. *Artificial Intelligence*, 221:36 – 45, 2015.
- [99] U. M. Erdem and M. E. Hasselmo. A goal-directed spatial navigation model using forward trajectory planning based on grid cells. *Eur J Neurosci*, 35(6):916–931, Mar 2012.
- [100] U. M. Erdem, M. J. Milford, and M. E. Hasselmo. A hierarchical model of goal directed navigation selects trajectories in a visual environment. *Neurobiology of Learning and Memory*, 117:109 – 121, 2015. Memory and decision making.
- [101] S. K. Esser, A. Andreopoulos, R. Appuswamy, P. Datta, D. Barch, A. Amir, J. Arthur, A. Cassidy, M. Flickner, P. Merolla, S. Chandra, N. Basilico, S. Carpin, T. Zimmerman, F. Zee, R. Alvarez-Icaza, J. A. Kusnitz, T. M. Wong, W. P. Risk, E. McQuinn, T. K. Nayak, R. Singh, and D. S. Modha. Cognitive computing systems: Algorithms and applications for networks of neurosynaptic cores. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10, Aug 2013.
- [102] D. R. Euston, A. J. Gruber, and B. L. McNaughton. The role of medial prefrontal cortex in memory and decision making. *Neuron*, 76(6):1057–1070, Dec 2012.
- [103] L. Fejes. Über die dichteste Kugellagerung. *Mathematische Zeitschrift*, 48(1):676–684, 1942.
- [104] D. E. Feldman. The spike-timing dependence of plasticity. *Neuron*, 75(4):556–571, 2012.
- [105] A. A. Fenton, H. Y. Kao, S. A. Neymotin, A. Olypher, Y. Vayntrub, W. W. Lytton, and N. Ludvig. Unmasking the CA1 ensemble place code by exposures to small and large environments: more place cells and multiple, irregularly arranged, and expanded place fields in the larger space. *J. Neurosci.*, 28(44):11250–11262, Oct 2008.
- [106] D. Ferrucci, A. Levas, S. Bagchi, D. Gondek, and E. T. Mueller. Watson: Beyond Jeopardy! *Artificial Intelligence*, 199:93 – 105, 2013.
- [107] D. M. Finch, N. L. Nowlin, and T. L. Babb. Demonstration of axonal projections of neurons in the rat hippocampus and subiculum by intracellular injection of HRP. *Brain Res.*, 271(2):201–216, Jul 1983.
- [108] R. A. Finkel and J. L. Bentley. Quad trees: A data structure for retrieval on composite keys. *Acta Informatica*, 4(1):1–9, 1974.

- [109] R. Fitzhugh. Impulses and Physiological States in Theoretical Models of Nerve Membrane. *Biophys. J.*, 1(6):445–466, Jul 1961.
- [110] M. O. Franz, B. Schölkopf, H. A. Mallot, and H. H. Bülthoff. Learning view graphs for robot navigation. In G. A. Bekey, editor, *Autonomous Agents*, pages 111–125. Springer US, Boston, MA, 1998.
- [111] M. Franzius, N. Wilbert, and L. Wiskott. Invariant object recognition with slow feature analysis. In V. Kůrková, R. Neruda, and J. Koutník, editors, *Artificial Neural Networks - ICANN 2008: 18th International Conference, Prague, Czech Republic, September 3-6, 2008, Proceedings, Part I*, pages 961–970, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- [112] T. Freund and G. Buzsáki. Interneurons of the hippocampus. *Hippocampus*, 6(4):347–470, 1996.
- [113] B. Fritzke. A growing neural gas network learns topologies. In *Advances in Neural Information Processing Systems 7*, pages 625–632. MIT Press, 1995.
- [114] E. C. Fuchs, A. Neitz, R. Pinna, S. Melzer, A. Caputi, and H. Monyer. Local and Distant Input Controlling Excitation in Layer ii of the Medial Entorhinal Cortex. *Neuron*, 89(1):194–208, 2017/04/07 2016.
- [115] E. Fuchs, A. Neitz, R. Pinna, S. Melzer, A. Caputi, and H. Monyer. Local and distant input controlling excitation in layer ii of the medial entorhinal cortex. *Neuron*, 89(1):194–208, 12 2015.
- [116] M. C. Fuhs and D. S. Touretzky. A spin glass model of path integration in rat medial entorhinal cortex. *J. Neurosci.*, 26(16):4266–4276, Apr 2006.
- [117] M. Fyhn, T. Hafting, A. Treves, M.-B. Moser, and E. I. Moser. Hippocampal remapping and grid realignment in entorhinal cortex. *Nature*, 446(7132):190–194, Mar 2007.
- [118] M. Fyhn, T. Solstad, and T. Hafting. Entorhinal grid cells and the neural basis of navigation. In *Hippocampal Place Fields*, pages 237–252. Oxford University Press (OUP), mar 2008.
- [119] E. Gamma. *Design patterns : elements of reusable object-oriented software*. Addison-Wesley, Reading, Mass, 1995.
- [120] M. A. Georgeson, K. A. May, T. C. A. Freeman, and G. S. Hesse. From filters to features: Scale–space analysis of edge and blur coding in human vision. *Journal of Vision*, 7(13):7, 2007.
- [121] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski. *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. Cambridge University Press, 2014.
- [122] L. M. Giocomo, M.-B. Moser, and E. I. Moser. Computational models of grid cells. *Neuron*, 71(4):589 – 603, 2011.

- [123] M. F. Glasser, T. S. Coalson, E. C. Robinson, C. D. Hacker, J. Harwell, E. Yacoub, K. Ugurbil, J. Andersson, C. F. Beckmann, M. Jenkinson, S. M. Smith, and D. C. Van Essen. A multi-modal parcellation of human cerebral cortex. *Nature*, 536(7615):171–178, Aug 2016. Article.
- [124] S. Gluth, T. Sommer, J. Rieskamp, and C. Büchel. Effective connectivity between hippocampus and ventromedial prefrontal cortex controls preferential choices from memory. *Neuron*, 86(4):1078–1090, 2017/04/06 2015.
- [125] A. Gorchetchnikov and S. Grossberg. Space, time and learning in the hippocampus: how fine spatial and temporal scales are expanded into population codes for behavioral control. *Neural Netw*, 20(2):182–193, Mar 2007.
- [126] K. M. Gothard, W. E. Skaggs, and B. L. McNaughton. Dynamics of mismatch correction in the hippocampal ensemble code for space: Interaction between path integration and environmental cues. *Journal of Neuroscience*, 16(24):8027–8040, 1996.
- [127] B. D. Gouveia, D. Portugal, D. C. Silva, and L. Marques. Computation Sharing in Distributed Robotic Systems: A Case Study on SLAM. *IEEE Transactions on Automation Science and Engineering*, 12(2):410–422, April 2015.
- [128] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):855–868, May 2009.
- [129] A. Graves and J. Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *Proceedings of the 21st International Conference on Neural Information Processing Systems, NIPS'08*, pages 545–552, USA, 2008. Curran Associates Inc.
- [130] C. M. Gray and W. Singer. Stimulus-specific neuronal oscillations in orientation columns of cat visual cortex. *Proc. Natl. Acad. Sci. U.S.A.*, 86(5):1698–1702, Mar 1989.
- [131] R. M. Grieves and K. J. Jeffery. The representation of space in the brain. *Behavioural Processes*, 135:113 – 131, 2017.
- [132] R. M. Grieves and P. A. Dudchenko. Cognitive maps and spatial inference in animals: Rats fail to take a novel shortcut, but can take a previously experienced one. *Learning and Motivation*, 44(2):81 – 92, 2013.
- [133] S. Grossberg and P. K. Pilly. Coordinated learning of grid cell and place cell spatial and temporal properties: multiple scales, attention and oscillations. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 369(1635):20120524, Feb 2014.
- [134] T. Hafting, M. Fyhn, T. Bonnevie, M.-B. Moser, and E. I. Moser. Hippocampus-independent phase precession in entorhinal grid cells. *Nature*, 453(7199):1248–1252, Jun 2008.

- [135] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, and E. I. Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806, Aug 2005.
- [136] M. Hagiya. Discrete state transition systems on continuous space-time: A theoretical model for amorphous computing. In C. S. Calude, M. J. Dinneen, G. Păun, M. J. Pérez-Jimenez, and G. Rozenberg, editors, *Unconventional Computation: 4th International Conference, UC 2005, Sevilla, Spain, October 3 – 7, 2005. Proceedings*, pages 117–129, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.
- [137] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems, Science, and Cybernetics*, SSC-4(2):100–107, 1968.
- [138] P. E. Hart, N. J. Nilsson, and B. Raphael. Correction to "a formal basis for the heuristic determination of minimum cost paths". *SIGART Bull.*, 37:28–29, December 1972.
- [139] C. D. Harvey, F. Collman, D. A. Dombeck, and D. W. Tank. Intracellular dynamics of hippocampal place cells during virtual navigation. *Nature*, 461(7266):941–946, oct 2009.
- [140] M. E. Hasselmo. What is the function of hippocampal theta rhythm?—Linking behavioral data to phasic properties of field potential and unit recording data. *Hippocampus*, 15(7):936–949, 2005.
- [141] M. E. Hasselmo. Neuronal rebound spiking, resonance frequency and theta cycle skipping may contribute to grid cell firing in medial entorhinal cortex. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 369(1635):20120523, Feb 2014.
- [142] M. Hattori and Y. Kobayashi. A hippocampal model for episodic memory using neurogenesis and asymmetric STDP. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 5189–5193, July 2016.
- [143] H. Hayashi and J. Igarashi. LTD windows of the STDP learning rule and synaptic connections having a large transmission delay enable robust sequence learning amid background noise. *Cogn Neurodyn*, 3(2):119–130, Jun 2009.
- [144] S. O. Haykin. *Neural Networks and Learning Machines*. Pearson, 2011.
- [145] D. O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. Wiley, New York, new ed edition, June 1949.
- [146] J. Hirel, P. Gaussier, M. Quoy, and J.-P. Banquet. Why and how hippocampal transition cells can be used in reinforcement learning. In S. Doncieux, B. Girard, A. Guillot, J. Hallam, J.-A. Meyer, and J.-B. Mouret, editors, *From Animals to Animats 11: 11th International Conference on Simulation of Adaptive Behavior, SAB 2010, Paris - Clos Lucé, France, August 25-28, 2010. Proceedings*, pages 359–369, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

- [147] C. A. R. Hoare. Communicating sequential processes. *Commun. ACM*, 21(8):666–677, August 1978.
- [148] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol. (Lond.)*, 117(4):500–544, Aug 1952.
- [149] K. Hoffman, A. Babichev, and Y. Dabaghian. A model of topological mapping of space in bat hippocampus. *Hippocampus*, 26(10):1345–1353, 2016.
- [150] M. Holzer, F. Schulz, and D. Wagner. Engineering multilevel overlay graphs for shortest-path queries. *J. Exp. Algorithmics*, 13:5:2.5–5:2.26, February 2009.
- [151] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. U.S.A.*, 79(8):2554–2558, Apr 1982.
- [152] M. A. W. Houtsma, P. M. G. Apers, and S. Ceri. Distributed transitive closure computations: The disconnection set approach. In *Proceedings of the 16th International Conference on Very Large Data Bases, VLDB '90*, pages 335–346, San Francisco, CA, USA, 1990. Morgan Kaufmann Publishers Inc.
- [153] A. Howard. Multi-robot mapping using manifold representations. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, volume 4, pages 4198–4203 Vol.4, April 2004.
- [154] A. Howard. Multi-robot simultaneous localization and mapping using particle filters. *The International Journal of Robotics Research*, 25(12):1243–1256, 2006.
- [155] Y.-W. Huang, N. Jing, and E. Rundensteiner. A hierarchical path view model for path finding in intelligent transportation systems. *GeoInformatica*, 1(2):125–159, 1997.
- [156] Y.-W. Huang, Y. wu Huangy, N. Jing, and E. A. Rundensteiner. Hierarchical path views: A model based on fragmentation and transportation road types, 1995.
- [157] D. Hubel and T. Wiesel. Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160:106–154, 1962.
- [158] G. M. Hunter and K. Steiglitz. Operations on images using quad trees. *IEEE Trans Pattern Anal Mach Intell*, 1(2):145–153, Feb 1979.
- [159] P. M. Iannaccone and M. Khokha. *Fractal Geometry in Biological Systems: An Analytical Approach*. CRC Press, 1996.
- [160] G. Indiveri, B. Linares-Barranco, T. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. SAÏGHI, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen. Neuromorphic silicon neuron circuits. *Frontiers in Neuroscience*, 5:73, 2011.

- [161] H. T. Ito. Prefrontal–hippocampal interactions for spatial navigation. *Neuroscience Research*, pages –, 2017.
- [162] E. M. Izhikevich. *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting (Computational Neuroscience)*. The MIT Press, 2006.
- [163] E. M. Izhikevich and N. S. Desai. Relating STDP to BCM. *Neural computation*, 15(7):1511–1523, July 2003.
- [164] S. P. Jadhav, C. Kemere, P. W. German, and L. M. Frank. Awake hippocampal sharp-wave ripples support spatial memory. *Science*, 336(6087):1454–1458, 2012.
- [165] M. M. Jankowski and S. M. O’Mara. Dynamics of place, boundary and object encoding in rat anterior claustrum. *Front Behav Neurosci*, 9:250, 2015.
- [166] M. M. Jankowski, J. Passecker, M. N. Islam, S. Vann, J. T. Erichsen, J. P. Aggleton, and S. M. O’Mara. Evidence for spatially-responsive neurons in the rostral thalamus. *Front Behav Neurosci*, 9:256, 2015.
- [167] L. E. Jarrard. On the role of the hippocampus in learning and memory in the rat. *Behav. Neural Biol.*, 60(1):9–26, Jul 1993.
- [168] M. Jazayeri and J. A. Movshon. Optimal representation of sensory information by neural populations. *Nat Neurosci*, 9(5):690–696, May 2006.
- [169] A. Jeewajee, C. Barry, V. Douchamps, D. Manson, C. Lever, and N. Burgess. Theta phase precession of grid and place cell firing in open environments. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 369(1635), 2013.
- [170] N. Jing, Y. W. Huang, and E. A. Rundensteiner. Hierarchical encoded path views for path query processing: An optimal model and its performance evaluation. *Knowledge and Data Engineering*, 10(3):409–432, 1998.
- [171] N. Jing, Y.-W. Huang, and E. A. Rundensteiner. Hierarchical optimization of optimal path finding for transportation applications. In *Proceedings of the Fifth International Conference on Information and Knowledge Management, CIKM ’96*, pages 261–268, New York, NY, USA, 1996. ACM.
- [172] S. H. Jo, T. Chang, I. Ebong, B. B. Bhadviya, P. Mazumder, and W. Lu. Nanoscale memristor device as synapse in neuromorphic systems. *Nano letters*, 10(4):1297–1301, 2010.
- [173] M. W. Jones and M. A. Wilson. Theta rhythms coordinate hippocampal-prefrontal interactions in a spatial memory task. *PLOS Biology*, 3(12), 11 2005.
- [174] S. Jones and J. L. Yakel. Inhibitory interneurons in hippocampus. *Cell Biochemistry and Biophysics*, 31(2):207–218, 1999.

- [175] S. Jung and S. Pramanik. An efficient path computation model for hierarchically structured topographical road maps. *IEEE Transactions on Knowledge and Data Engineering*, 14(5):1029–1046, 2002.
- [176] M. Kaiser, C. C. Hilgetag, and R. Kotter. Hierarchy and dynamics of neural networks. *Front Neuroinform*, 4, 2010.
- [177] E. R. Kandel, J. H. Schwartz, T. M. Jessell, and S. Mack, editors. *Principles of neural science*. McGraw-Hill Medical, New York, Chicago, San Francisco, 2013.
- [178] J. Kerdels and G. Peters. A computational model of grid cells based on dendritic self-organized learning. In *Proceedings of the 5th International Joint Conference on Computational Intelligence (IJCCI 2013)*, pages 420–429, 2013.
- [179] T. Kitanishi, H. T. Ito, Y. Hayashi, Y. Shinohara, K. Mizuseki, and T. Hikida. Network mechanisms of hippocampal laterality, place coding, and goal-directed navigation. *The Journal of Physiological Sciences*, pages 1–12, 2016.
- [180] A. Knoblauch. Zip nets: Efficient associative computation with binary synapses. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, July 2010.
- [181] A. Knoblauch, G. Palm, and F. T. Sommer. Memory capacities for synaptic and structural plasticity. *Neural Comput*, 22(2):289–341, Feb 2010.
- [182] A. Knoblauch. Synchronization and pattern separation in spiking associative memories and visual cortical areas, 2004.
- [183] A. Knoblauch. Neural associative memory for brain modeling and information retrieval. *Information Processing Letters*, 95(6):537 – 544, 2005. Applications of Spiking Neural Networks.
- [184] C. Knoblock. Search reduction in hierarchical problem solving. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91)*”, volume 2, pages 686–691, Anaheim, California, USA, 1991.
- [185] D. E. Knuth. *The Art of Computer Programming, Volume 3: (2nd Ed.) Sorting and Searching*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1998.
- [186] C. Koch. *Biophysics of Computation: Information Processing in Single Neurons (Computational Neuroscience Series)*. Oxford University Press, Inc., New York, NY, USA, 2004.
- [187] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50(5):363–370, 1984.
- [188] R. A. Koene and M. E. Hasselmo. Reversed and forward buffering of behavioral spike sequences enables retrospective and prospective retrieval in hippocampal regions CA3 and CA1. *Neural Netw*, 21(2-3):276–288, 2008.

- [189] J. Koenig, A. N. Linder, J. K. Leutgeb, and S. Leutgeb. The spatial periodicity of grid cells is not sustained during reduced theta oscillations. *Science*, 332(6029):592–595, 2011.
- [190] C. Kohler. Intrinsic projections of the retrohippocampal region in the rat brain. I. The subicular complex. *J. Comp. Neurol.*, 236(4):504–522, Jun 1985.
- [191] T. Kohonen. Self-organized formation of topologically correct feature maps. In J. A. Anderson and E. Rosenfeld, editors, *Neurocomputing: Foundations of Research*, pages 509–521. MIT Press, Cambridge, MA, USA, 1988.
- [192] B. Kolb. Functions of the frontal cortex of the rat: a comparative review. *Brain Res.*, 320(1):65–98, Nov 1984.
- [193] B. J. Kraus, M. P. Brandon, R. J. I. Robinson, M. A. Connerney, M. E. Hasselmo, and H. Eichenbaum. During running in place, grid cells integrate elapsed time and distance run. *Neuron*, 88(3):578–589, 2017/02/10 2015.
- [194] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, NIPS’12, pages 1097–1105, USA, 2012. Curran Associates Inc.
- [195] E. Kropff, J. E. Carmichael, M.-B. Moser, and E. I. Moser. Speed cells in the medial entorhinal cortex. *Nature*, 523(7561):419–424, Jul 2015. Article.
- [196] E. Kropff and A. Treves. The emergence of grid cells: Intelligent design or just adaptation. *Hippocampus*, 18(12):1256–1269, 2008.
- [197] J. Krupic, M. Bauza, S. Burton, C. Lever, and J. O’Keefe. How environment geometry affects grid cell symmetry and what we can learn from it. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 369(1635):20130188, Feb 2014.
- [198] J. Krupic, M. Bauza, S. Burton, C. Barry, and J. O’Keefe. Grid cell symmetry is shaped by environmental geometry. *Nature*, 518(7538):232–235, Feb 2015. Letter.
- [199] J. Krupic, M. Bauza, S. Burton, and J. O’Keefe. Framing the grid: effect of boundaries on grid cells and navigation. *The Journal of Physiology*, 594(22):6489–6499, 2016.
- [200] J. L. Kubie and A. A. Fenton. Linear look-ahead in conjunctive cells: an entorhinal mechanism for vector-based navigation. *Front Neural Circuits*, 6:20, 2012.
- [201] D. M. Kullmann. Interneuron networks in the hippocampus. *Curr. Opin. Neurobiol.*, 21(5):709–716, Oct 2011.
- [202] H. R. Kunsch, E. Agrell, and F. A. Hamprecht. Optimal lattices for sampling. *IEEE Transactions on Information Theory*, 51(2):634–647, Feb 2005.

- [203] L. Lamport. Time, clocks, and the ordering of events in a distributed system. *Commun. ACM*, 21(7):558–565, July 1978.
- [204] R. F. Langston, J. A. Ainge, J. J. Couey, C. B. Canto, T. L. Bjerknes, M. P. Witter, E. I. Moser, and M. B. Moser. Development of the spatial representation system in the rat. *Science*, 328(5985):1576–1580, Jun 2010.
- [205] C. Le Duigou, J. Simonnet, M. T. Telenczuk, D. Fricker, and R. Miles. Recurrent synapses and circuits in the CA3 region of the hippocampus: an associative network. *Front Cell Neurosci*, 7:262, Jan 2014.
- [206] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.
- [207] I. Lee, D. Yoganarasimha, G. Rao, and J. J. Knierim. Comparison of population coherence of place cells in hippocampal subfields ca1 and ca3. *Nature*, 430(6998):456–459, Jul 2004.
- [208] T. S. Lee and D. Mumford. Hierarchical Bayesian inference in the visual cortex. *J Opt Soc Am A Opt Image Sci Vis*, 20(7):1434–1448, Jul 2003.
- [209] J. Leech and N. J. A. Sloane. *Sphere Packing and Error-Correcting Codes*, pages 136–156. Springer New York, New York, NY, 1999.
- [210] J. K. Leutgeb, S. Leutgeb, M. B. Moser, and E. I. Moser. Pattern separation in the dentate gyrus and CA3 of the hippocampus. *Science*, 315(5814):961–966, Feb 2007.
- [211] C. Lever, S. Burton, A. Jeewajee, J. O’Keefe, and N. Burgess. Boundary vector cells in the subiculum of the hippocampal formation. *J. Neurosci.*, 29(31):9771–9777, Aug 2009.
- [212] M. Li, J. Liu, and J. Z. Tsien. Theory of connectivity: Nature and nurture of cell assemblies and cognitive computation. *Front Neural Circuits*, 10:34, Apr 2016.
- [213] T. Lindeberg. Scale-space theory: A basic tool for analyzing structures at different scales. *Journal of Applied Statistics*, 21:225–270, 1994.
- [214] T. Lindeberg. A computational theory of visual receptive fields. *Biol Cybern*, 107(6):589–635, Dec 2013.
- [215] T. Lindeberg. *Scale-Space Theory in Computer Vision (The Springer International Series in Engineering and Computer Science)*. Springer, 2010.
- [216] T. Lindeberg. Image matching using generalized scale-space interest points. *Journal of Mathematical Imaging and Vision*, 52(1):3–36, 2015.
- [217] M. Llofriu, G. Tejera, M. Contreras, T. Pelc, J. M. Fellous, and A. Weitzenfeld. Goal-oriented robot navigation learning using a multi-scale space representation. *Neural Netw*, 72:62–74, Dec 2015.

- [218] K. Lodaya, R. Parikh, R. Ramanujam, and P. Thiagarajan. A logical study of distributed transition systems. *Information and Computation*, 119(1):91 – 118, 1995.
- [219] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2):91–110, November 2004.
- [220] D. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*. IEEE, 1999.
- [221] E. V. Lubenov and A. G. Siapas. Hippocampal theta oscillations are travelling waves. *Nature*, 459(7246):534–539, May 2009.
- [222] J. Lücke and J. D. Bouecke. Dynamics of cortical columns – self-organization of receptive fields. In *Proceedings of the 15th International Conference on Artificial Neural Networks: Biological Inspirations - Volume Part I, ICANN'05*, pages 31–37, Berlin, Heidelberg, 2005. Springer-Verlag.
- [223] T. V. Maia. Reinforcement learning, conditioning, and the brain: Successes and challenges. *Cogn Affect Behav Neurosci*, 9(4):343–364, Dec 2009.
- [224] N. Maier, A. Draguhn, D. Schmitz, and M. Both. Fast network oscillations in the hippocampus. *e-Neuroforum*, 4(1):1–10, 2013.
- [225] J. R. Manns and H. Eichenbaum. A cognitive map for object memory in the hippocampus. *Learn. Mem.*, 16(10):616–624, Oct 2009.
- [226] H. Markram, W. Gerstner, and P. J. Sjöström. Spike-timing-dependent plasticity: a comprehensive overview. *Front Synaptic Neurosci*, 4:2, 2012.
- [227] E. Marozzi, L. L. Ginzberg, A. Alenda, and K. J. Jeffery. Purely Translational Realignment in Grid Cell Firing Patterns Following Nonmetric Context Change. *Cereb. Cortex*, 25(11):4619–4627, Nov 2015.
- [228] E. Marozzi and K. J. Jeffery. Place, space and memory cells. *Current Biology*, 22(22):R939 – R942, 2012.
- [229] D. Marr. Simple memory: a theory for archicortex. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 262(841):23–81, Jul 1971.
- [230] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., New York, NY, USA, 1982.
- [231] T. Martinetz, K. Schulten, et al. A "neural-gas" network learns topologies. *Artificial Neural Networks*, pages 397–402, 1991.
- [232] T. Masquelier and S. J. Thorpe. Unsupervised learning of visual features through spike timing dependent plasticity. *PLOS Computational Biology*, 3(2):1–11, 02 2007.

- [233] A. Mathis, A. V. M. Herz, and M. Stemmler. Optimal population codes for space: Grid cells outperform place cells. *Neural Computation*, 24(9):2280–2317, May 2012.
- [234] J. Maue, P. Sanders, and D. Matijevic. Goal-directed shortest-path queries using precomputed cluster distances. *Journal of Experimental Algorithmics (JEA)*, 14:2, 2009.
- [235] A. P. Maurer and B. L. McNaughton. Network and intrinsic cellular mechanisms underlying theta phase precession of hippocampal neurons. *Trends in Neurosciences*, 30(7):325–333, jul 2007.
- [236] M. May-Britt. Place cells and grid cells in the developing rat brain. *Frontiers in Behavioral Neuroscience*, 3, 2009.
- [237] M. Mayford, S. A. Siegelbaum, and E. R. Kandel. Synapses and memory storage. *Cold Spring Harbor Perspectives in Biology*, 4(6):a005751–a005751, apr 2012.
- [238] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.
- [239] T. J. McHugh, M. W. Jones, J. J. Quinn, N. Balthasar, R. Coppari, J. K. Elmquist, B. B. Lowell, M. S. Fanselow, M. A. Wilson, and S. Tonegawa. Dentate gyrus NMDA receptors mediate rapid pattern separation in the hippocampal network. *Science*, 317(5834):94–99, Jul 2007.
- [240] D. McNamee, D. M. Wolpert, and M. Lengyel. Efficient state-space modularization for planning: theory, behavioral and neural signatures. In *Advances in Neural Information Processing Systems*, pages 4511–4519, 2016.
- [241] B. L. McNaughton and R. G. M. Morris. Hippocampal synaptic enhancement and information storage within a distributed memory system. *Trends in Neurosciences*, 10(10):408–415, 2017/03/06 1987.
- [242] M. R. Mehta, A. K. Lee, and M. A. Wilson. Role of experience and oscillations in transforming a rate code into a temporal code. *Nature*, 417(6890):741–746, Jun 2002.
- [243] M. R. Mehta and B. L. McNaughton. Expansion and shift of hippocampal place fields: Evidence for synaptic potentiation during behavior. In J. M. Bower, editor, *Computational Neuroscience: Trends in Research, 1997*, pages 741–745. Springer US, Boston, MA, 1997.
- [244] J. d. R. Millán. Robot navigation. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks: The Second Edition*. The MIT Press, 0 2002.
- [245] R. K. Mishra, S. Kim, S. J. Guzman, and P. Jonas. Symmetric spike timing-dependent plasticity at ca3-ca3 synapses optimizes storage and recall in autoassociative networks. *Nature Communications*, 7:11552 EP –, May 2016. Article.

- [246] K. Mizuseki, A. Sirota, E. Pastalkova, and G. Buzsaki. Theta oscillations provide temporal windows for local circuit computation in the entorhinal-hippocampal loop. *Neuron*, 64(2):267–280, Oct 2009.
- [247] M. A. Moita, S. Rosis, Y. Zhou, J. E. LeDoux, and H. T. Blair. Hippocampal place cells acquire location-specific responses to the conditioned stimulus during auditory fear conditioning. *Neuron*, 37(3):485–497, Feb 2003.
- [248] J. D. Monaco and L. F. Abbott. Modular realignment of entorhinal grid cell activity as a basis for hippocampal remapping. *Journal of Neuroscience*, 31(25):9414–9425, jun 2011.
- [249] R. Morris. Developments of a water-maze procedure for studying spatial learning in the rat. *J. Neurosci. Methods*, 11(1):47–60, May 1984.
- [250] R. G. Morris. Long-term potentiation and memory. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 358(1432):643–647, Apr 2003.
- [251] R. G. Morris, P. Garrud, J. N. Rawlins, and J. O’Keefe. Place navigation impaired in rats with hippocampal lesions. *Nature*, 297(5868):681–683, Jun 1982.
- [252] R. G. Morris and D. O. Hebb. D.O. Hebb: The Organization of Behavior, Wiley: New York; 1949. *Brain Res. Bull.*, 50(5-6):437, 1999.
- [253] E. I. Moser, E. Kropff, and M. B. Moser. Place cells, grid cells, and the brain’s spatial representation system. *Annu. Rev. Neurosci.*, 31:69–89, 2008.
- [254] E. I. Moser and M. B. Moser. A metric for space. *Hippocampus*, 18(12):1142–1156, 2008.
- [255] E. I. Moser and M.-B. Moser. Grid cells and neural coding in high-end cortices. *Neuron*, 80(3):765 – 774, 2013.
- [256] E. I. Moser, Y. Roudi, M. P. Witter, C. Kentros, T. Bonhoeffer, and M.-B. Moser. Grid cells and cortical representation. *Nat Rev Neurosci*, 15(7):466–481, Jul 2014. Review.
- [257] M. B. Moser, D. C. Rowland, and E. I. Moser. Place cells, grid cells, and memory. *Cold Spring Harb Perspect Biol*, 7(2):a021808, Feb 2015.
- [258] M. Mulas, N. Waniek, and J. Conradt. Hebbian plasticity realigns grid cell activity with external sensory cues in continuous attractor models. *Front Comput Neurosci*, 10:13, Feb 2016.
- [259] R. U. Muller, J. L. Kubie, and J. B. Ranck. Spatial firing patterns of hippocampal complex-spike cells in a fixed environment. *J. Neurosci.*, 7(7):1935–1950, Jul 1987.
- [260] K. Nakazawa, L. D. Sun, M. C. Quirk, L. Rondi-Reig, M. A. Wilson, and S. Tonegawa. Hippocampal CA3 NMDA receptors are crucial for memory acquisition of one-time experience. *Neuron*, 38(2):305 – 315, 2003.

- [261] C. Nicolini and A. Bifone. Modular structure of brain functional networks: breaking the resolution limit by surprise. *Scientific Reports*, 6:19250 EP –, Jan 2016. Article.
- [262] J. E. Niven and S. B. Laughlin. Energy limitation as a selective pressure on the evolution of sensory systems. *J. Exp. Biol.*, 211(Pt 11):1792–1804, Jun 2008.
- [263] J.-R. Ohm. *Multimedia Signal Coding and Transmission*. Springer Publishing Company, Incorporated, 2015.
- [264] J. O’Keefe. A review of the hippocampal place cells. *Prog. Neurobiol.*, 13(4):419–439, 1979.
- [265] J. O’Keefe and J. Dostrovsky. The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, 34(1):171 – 175, 1971.
- [266] J. O’Keefe and N. Burgess. Geometric determinants of the place fields of hippocampal neurons. *Nature*, 381(6581):425–428, may 1996.
- [267] J. A. Orenstein. Multidimensional tries used for associative searching. *Information Processing Letters*, 14(4):150 – 157, 1982.
- [268] E. Painkras, L. A. Plana, J. Garside, S. Temple, S. Davidson, J. Pepper, D. Clark, C. Patterson, and S. Furber. SpiNNaker: A multi-core System-on-Chip for massively-parallel neural net simulation. In *Proceedings of the IEEE 2012 Custom Integrated Circuits Conference*, pages 1–4, Sept 2012.
- [269] G. Palm. On associative memory. *Biological Cybernetics*, 36(1):19–31, 1980.
- [270] G. Palm. *Neural Assemblies, an Alternative Approach to Artificial Intelligence*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1982.
- [271] G. Palm. On the information storage capacity of local learning rules. *Neural Comput.*, 4(5):703–711, September 1992.
- [272] G. Palm. *Novelty, Information and Surprise*. Springer, 2012.
- [273] G. Palm. Neural associative memories and sparse coding. *Neural Networks*, 37:165 – 171, 2013. Twenty-fifth Anniversary Commemorative Issue.
- [274] A. Pantazi, S. Woźniak, T. Tuma, and E. Eleftheriou. All-memristive neuro-morphic computing with level-tuned neurons. *Nanotechnology*, 27(35):355205, 2016.
- [275] D. Papo, J. M. Buldu, S. Boccaletti, and E. T. Bullmore. Complex network theory and the brain. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 369(1653), Oct 2014.
- [276] D. Papo, J. M. Buldú, and S. Boccaletti. Network theory in neuroscience. In D. Jaeger and R. Jung, editors, *Encyclopedia of Computational Neuroscience*, pages 1–21. Springer New York, New York, NY, 2013.

- [277] G. Papp, M. P. Witter, and A. Treves. The CA3 network as a memory store for spatial representations. *Learn. Mem.*, 14(11):732–744, Nov 2007.
- [278] E. Park, D. Dvorak, and A. A. Fenton. Ensemble place codes in hippocampus: Ca1, ca3, and dentate gyrus place cells have multiple place fields in large environments. *PLOS ONE*, 6(7):1–9, 07 2011.
- [279] E. Pastalkova, V. Itskov, A. Amarasingham, and G. Buzsáki. Internally generated cell assembly sequences in the rat hippocampus. *Science*, 321(5894):1322–1327, Sep 2008.
- [280] J. A. Pérez-Escobar, O. Kornienko, P. Latuske, L. Kohler, and K. Allen. Visual landmarks sharpen grid cell metric and confer context specificity to neurons of the medial entorhinal cortex. *eLife*, 5:e16937, jul 2016.
- [281] D. P. Petersen and D. Middleton. Sampling and reconstruction of wave-number-limited functions in n-dimensional euclidean spaces. *Information and Control*, 5(4):279 – 323, 1962.
- [282] B. E. Pfeiffer and D. J. Foster. Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, 497(7447):74–79, May 2013.
- [283] J.-P. Pfister and W. Gerstner. Triplets of spikes in a model of spike timing-dependent plasticity. *Journal of Neuroscience*, 26(38):9673–9682, 2006.
- [284] A. R. Preston and H. Eichenbaum. Interplay of hippocampus and prefrontal cortex in memory. *Current Biology*, 23(17):R764 – R773, 2013.
- [285] P. Preston-Ferrer, S. Coletta, M. Frey, and A. Burgalossi. Anatomical organization of presubicular head-direction circuits. *eLife*, 5:e14592, jun 2016.
- [286] F. Pulvermuller, M. Garagnani, and T. Wennekers. Thinking in circuits: toward neurobiological explanation in cognitive neuroscience. *Biol Cybern*, 108(5):573–593, Oct 2014.
- [287] G. J. Quirk, R. U. Muller, J. L. Kubie, and J. B. Ranck. The positional firing properties of medial entorhinal neurons: description and comparison with hippocampal place cells. *J. Neurosci.*, 12(5):1945–1963, May 1992.
- [288] J. F. Ramirez-Villegas, N. K. Logothetis, and M. Besserve. Diversity of sharp-wave-ripple LFP signatures reveals differentiated brain-wide dynamical events. *Proc. Natl. Acad. Sci. U.S.A.*, 112(46):E6379–6387, Nov 2015.
- [289] J. B. Ranck. Head-direction cells in the deep cell layers of dorsal presubiculum in freely moving rats. *Society for Neuroscience Abstracts*, 10, 1984.
- [290] J. Ranck. Head direction cells in the deep cell layer of dorsal presubiculum in freely moving rats. In C. V. E. G. Buzsáki, editor, *Electrical Activity of the Archicortex*, pages 217–220. Akademiai Kiado, Budapest, 1985.
- [291] F. Raudies and M. E. Hasselmo. Differences in Visual-Spatial Input May Underlie Different Compression Properties of Firing Fields for Grid Cell Modules in Medial Entorhinal Cortex. *PLoS Comput. Biol.*, 11(11):e1004596, Nov 2015.

- [292] F. Raudies, E. Mingolla, and M. E. Hasselmo. Modeling the influence of optic flow on grid cell firing in the absence of other cues. *J Comput Neurosci*, 33(3):475–493, Dec 2012.
- [293] F. Raudies, M. P. Brandon, G. W. Chapman, and M. E. Hasselmo. Head direction is coded more strongly than movement direction in a population of entorhinal neurons. *Brain Research*, 1621:355 – 367, 2015. Brain and Memory: Old Arguments and New Perspectives.
- [294] F. Raudies and M. E. Hasselmo. Modeling boundary vector cell firing given optic flow as a cue. *PLOS Computational Biology*, 8(6):1–17, 06 2012.
- [295] F. Raudies, F. Raudies, E. Mingolla, and M. Hasselmo. Does optic flow explain the firing of grid cells? *Nature Precedings*, mar 2011.
- [296] E. Remolina and B. Kuipers. Towards a general theory of topological maps. *Artificial Intelligence*, 152(1):47 – 104, 2004.
- [297] P. Rodrigues, J. F. Costa, and H. T. Siegelmann. Verifying properties of neural networks. In J. Mira and A. Prieto, editors, *Connectionist Models of Neurons, Learning Processes, and Artificial Intelligence: 6th International Work-Conference on Artificial and Natural Neural Networks, IWANN 2001 Granada, Spain, June 13–15, 2001 Proceedings, Part 1*, pages 158–165. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.
- [298] P. R. Roelfsema. Solutions for the binding problem. *Z. Naturforsch., C, J. Biosci.*, 53(7-8):691–715, 1998.
- [299] E. T. Rolls. An attractor network in the hippocampus: theory and neurophysiology. *Learn. Mem.*, 14(11):714–731, Nov 2007.
- [300] E. T. Rolls. Functions of the primate hippocampus in spatial and nonspatial memory. *Hippocampus*, 1(3):258–261, 1991.
- [301] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.
- [302] L. Roux, B. Hu, R. Eichler, E. Stark, and G. Buzsaki. Sharp wave ripples during learning stabilize the hippocampal spatial map. *Nat Neurosci*, advance online publication, Apr 2017. Article.
- [303] D. C. Rowland, Y. Roudi, M.-B. Moser, and E. I. Moser. Ten years of grid cells. *Annual Review of Neuroscience*, 39(1):19–40, jul 2016.
- [304] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. In J. A. Anderson and E. Rosenfeld, editors, *Neurocomputing: Foundations of Research*, pages 696–699. MIT Press, Cambridge, MA, USA, 1988.
- [305] D. E. Rumelhart, J. L. McClelland, and T. P. R. Group, editors. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations*. MIT Press, Cambridge, MA, USA, 1986.

- [306] D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, editors. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 2: Psychological and Biological Models*. MIT Press, Cambridge, MA, USA, 1986.
- [307] S. Saeedi, M. Trentini, M. Seto, and H. Li. Multiple-robot simultaneous localization and mapping: A review. *Journal of Field Robotics*, 33(1):3–46, 2016.
- [308] P. A. Salin, M. Scanziani, R. C. Malenka, and R. A. Nicoll. Distinct short-term plasticity at two excitatory synapses in the hippocampus. *Proc. Natl. Acad. Sci. U.S.A.*, 93(23):13304–13309, Nov 1996.
- [309] D. M. Salz, Z. Tiganj, S. Khasnabish, A. Kohley, D. Sheehan, M. W. Howard, and H. Eichenbaum. Time Cells in Hippocampal Area CA3. *J. Neurosci.*, 36(28):7476–7484, Jul 2016.
- [310] H. Samet. An overview of quadtrees, octrees, and related hierarchical data structures. In R. A. Earnshaw, editor, *Theoretical Foundations of Computer Graphics and CAD*, pages 51–68. Springer Berlin Heidelberg, Berlin, Heidelberg, 1988.
- [311] I. W. Sandberg, J. T. Lo, C. L. Fancourt, J. C. Principe, S. Katagiri, and S. Haykin. *Nonlinear Dynamical Systems: Feedforward Neural Network Perspectives*. Wiley-Interscience, 2001.
- [312] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. One-shot learning with memory-augmented neural networks, 2016.
- [313] F. Sargolini, M. Fyhn, T. Hafting, B. L. McNaughton, M. P. Witter, M. B. Moser, and E. I. Moser. Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, 312(5774):758–762, May 2006.
- [314] E. Save, A. Cressant, C. Thinus-Blanc, and B. Poucet. Spatial firing of hippocampal place cells in blind rats. *J. Neurosci.*, 18(5):1818–1826, Mar 1998.
- [315] F. Savelli, D. Yoganarasimha, and J. J. Knierim. Influence of boundary removal on the spatial representations of the medial entorhinal cortex. *Hippocampus*, 18(12):1270–1282, 2008.
- [316] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85 – 117, 2015.
- [317] C. Schmidt-Hieber and M. Hausser. How to build a grid cell. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, 369(1635):20120520, Feb 2014.
- [318] F. Schulz, D. Wagner, and C. Zaroliagis. Using multi-level graphs for timetable information in railway systems. In D. M. Mount and C. Stein, editors, *Algorithm Engineering and Experiments: 4th International Workshop, ALENEX 2002 San Francisco, CA, USA, January 4–5, 2002 Revised Papers*, pages 43–59, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

- [319] W. B. Scoville and B. Milner. Loss of recent memory after bilateral hippocampal lesions. *J Neurol Neurosurg Psychiatry*, 20(1):11–21, Feb 1957.
- [320] M. Segal. Dendritic spines and long-term plasticity. *Nat Rev Neurosci*, 6(4):277–284, Apr 2005.
- [321] P. E. Sharp. Computer simulation of hippocampal place cells. *Psychobiology*, 19(2):103–115, 1991.
- [322] C. F. Shay, M. Ferrante, G. W. Chapman, and M. E. Hasselmo. Rebound spiking in layer II medial entorhinal cortex stellate cells: Possible mechanism of grid cell function. *Neurobiol Learn Mem*, 129:83–98, Mar 2016.
- [323] G. M. Sheperd. Introduction to Synaptic Circuits. In *The Synaptic Organization of the Brain*, pages 1–38. Oxford University Press, Jan 2004.
- [324] O. Shipston-Sharman, L. Solanka, and M. F. Nolan. Continuous attractor network models of grid cell firing based on excitatory-inhibitory interactions. *J. Physiol. (Lond.)*, 594(22):6547–6557, Nov 2016.
- [325] B. Si, E. Kropff, and A. Treves. Grid alignment in entorhinal cortex. *Biological Cybernetics*, 106(8):483–506, 2012.
- [326] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, Jan 2016. Article.
- [327] R. G. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun, and H. L. S. Younes. Coordination for multi-robot exploration and mapping. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 852–858. AAAI Press, 2000.
- [328] W. E. Skaggs, B. L. McNaughton, M. A. Wilson, and C. A. Barnes. Theta phase precession in hippocampal neuronal populations and the compression of temporal sequences. *Hippocampus*, 6(2):149–172, 1996.
- [329] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*, pages 167–193. Springer, 1990.
- [330] T. Solstad, E. I. Moser, and G. T. Einevoll. From grid cells to place cells: a mathematical model. *Hippocampus*, 16(12):1026–1031, 2006.
- [331] T. Solstad, C. N. Boccara, E. Kropff, M.-B. Moser, and E. I. Moser. Representation of geometric borders in the entorhinal cortex. *Science*, 322(5909):1865–1868, 2008.
- [332] T. Solstad, H. N. Yousif, and T. J. Sejnowski. Place cell rate remapping by ca3 recurrent collaterals. *PLOS Computational Biology*, 10(6):1–10, 06 2014.

- [333] O. Sporns. Graph theory methods for the analysis of neural connectivity patterns. In R. Kötter, editor, *Neuroscience Databases: A Practical Guide*, pages 171–185. Springer US, Boston, MA, 2003.
- [334] S. Sreenivasan and I. Fiete. Grid cells generate an analog error-correcting code for singularly precise neural computation. *Nat Neurosci*, 14(10):1330–1337, Oct 2011. Article.
- [335] M. Stemmler, A. Mathis, and A. V. M. Herz. Connecting multiple spatial scales to decode the population activity of grid cells. *Science Advances*, 1(11), 2015.
- [336] H. Stensola, T. Stensola, T. Solstad, K. Froland, M.-B. Moser, and E. I. Moser. The entorhinal grid map is discretized. *Nature*, 492(7427):72–78, December 2012.
- [337] T. Stensola, H. Stensola, M.-B. Moser, and E. I. Moser. Shearing-induced asymmetry in entorhinal grid cells. *Nature*, 518(7538):207–212, Feb 2015. Article.
- [338] A. Stepanyuk. Self-organization of grid fields under supervision of place cells in a neuron model with associative plasticity. *Biologically Inspired Cognitive Architectures*, 13:48 – 62, 2015.
- [339] M. L. Sutter. Shapes and level tolerances of frequency tuning curves in primary auditory cortex: quantitative measures and population codes. *J. Neurophysiol.*, 84(2):1012–1025, Aug 2000.
- [340] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning series)*. A Bradford Book, 1998.
- [341] H. M. Tan, T. J. Wills, and F. Cacucci. The development of spatial and memory circuits in the rat. *Wiley Interdisciplinary Reviews: Cognitive Science*, dec 2016.
- [342] J. S. Taube. The head direction signal: origins and sensory-motor integration. *Annu. Rev. Neurosci.*, 30:181–207, 2007.
- [343] J. Taube, R. Muller, and J. Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *Journal of Neuroscience*, 10(2):420–435, 1990.
- [344] H. H. Teh. *Neural Logic Networks: A New Class of Neural Networks*. World Scientific Publishing Company, 1995.
- [345] W. Thomas. Automata theory and infinite transition systems, 2006.
- [346] L. T. Thompson and P. J. Best. Long-term stability of the place-field activity of single units recorded from the dorsal hippocampus of freely behaving rats. *Brain Res.*, 509(2):299–308, Feb 1990.

- [347] S. Thorpe, A. Delorme, and R. Van Rullen. Spike-based strategies for rapid processing. *Neural Netw*, 14(6-7):715–725, 2001.
- [348] S. Thrun. Particle filters in robotics. In *in Proceedings of the 17th Annual Conference on Uncertainty in AI (UAI)*, 2002.
- [349] S. Thrun. Self-Driving Cars – An AI-Robotics Challenge. In D. Wilson and G. Sutcliffe, editors, *Proceedings of the Twentieth International Florida Artificial Intelligence Research Society Conference, May 7-9, 2007, Key West, Florida, USA.*, page 12. AAAI Press, 2007.
- [350] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.
- [351] S. Thrun and Y. Liu. Multi-robot SLAM with sparse extended information filters. In *Robotics Research. The Eleventh International Symposium*, pages 254–266. Springer, 2005.
- [352] S. Thrun and M. Montemerlo. The graph SLAM algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research*, 25(5-6):403–429, 2006.
- [353] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Stanley: The robot that won the darpa grand challenge: Research articles. *J. Robot. Syst.*, 23(9):661–692, September 2006.
- [354] E. C. Tolman. Cognitive maps in rats and men. *Psychological Review*, 55(4):189–208, 1948.
- [355] V. Topac. Efficient fuzzy search enabled hash map. In *4th International Workshop on Soft Computing Applications*, pages 39–44, July 2010.
- [356] A. Treves and E. T. Rolls. Computational analysis of the role of the hippocampus in memory. *Hippocampus*, 4(3):374–391, Jun 1994.
- [357] A. Treves, O. Miglino, and D. Parisi. Rats, nets, maps, and the emergence of place cells. *Psychobiology*, 20(1):1–8, 1992.
- [358] E. Tulving. Episodic and semantic memory 1. *Organization of Memory. London: Academic*, 381(4):382–404, 1972.
- [359] A. M. Turing. The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 237(641):37–72, 1952.
- [360] N. Ulanovsky and C. F. Moss. Hippocampal cellular and network activity in freely moving echolocating bats. *Nat. Neurosci.*, 10(2):224–233, Feb 2007.

- [361] J. Van Benthem and J. Bergstra. Logic of transition systems. *Journal of Logic, Language and Information*, 3(4):247–283, 1994.
- [362] M. A. A. van der Meer and A. D. Redish. Theta phase precession in rat ventral striatum links place and reward information. *Journal of Neuroscience*, 31(8):2843–2854, 2011.
- [363] R. Van Rullen and S. J. Thorpe. Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex. *Neural Comput*, 13(6):1255–1283, Jun 2001.
- [364] N. M. van Strien, N. L. Cappaert, and M. P. Witter. The anatomy of memory: an interactive overview of the parahippocampal-hippocampal network. *Nat. Rev. Neurosci.*, 10(4):272–282, Apr 2009.
- [365] Z. Varga, H. Jia, B. Sakmann, and A. Konnerth. Dendritic coding of multiple sensory inputs in single cortical neurons in vivo. *Proceedings of the National Academy of Sciences*, 108(37):15420–15425, aug 2011.
- [366] R. P. Vertes. Hippocampal theta rhythm: a tag for short-term memory. *Hippocampus*, 15(7):923–935, 2005.
- [367] M. Viazovska. The sphere packing problem in dimension 8. *Annals of Mathematics*, 185(3):991–1015, may 2017.
- [368] S. Vinga and J. S. Almeida. Rényi continuous entropy of DNA sequences. *J. Theor. Biol.*, 231(3):377–388, Dec 2004.
- [369] C. von der Malsburg. The what and why of binding: the modeler’s perspective. *Neuron*, 24(1):95–104, Sep 1999.
- [370] C. von der Malsburg and W. Schneider. A neural cocktail-party processor. *Biological Cybernetics*, 54(1):29–40, 1986.
- [371] N. Waniek, J. Biedermann, and J. Conradt. Cooperative SLAM on small mobile robots. In *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1810–1815, Dec 2015.
- [372] N. Waniek, E. Berzs, and J. Conradt. Data structures for locally distributed routing. *submitted*.
- [373] N. Waniek, M. Mulas, and J. Conradt. Grid cell realignment based on idiothetic head direction cues, 2014. Poster presented at the BCCN Conference.
- [374] J. P. Weber, B. K. Andrasfalvy, M. Polito, A. Mago, B. B. Ujfalussy, and J. K. Makara. Location-dependent synaptic plasticity rules by dendritic spine cooperativity. *Nat Commun*, 7:11380, Apr 2016.
- [375] X.-x. Wei, J. Prentice, and V. Balasubramanian. A principle of economy predicts the functional architecture of grid cells. *eLife*, 2015.
- [376] T. Wennekers and G. Palm. On the relation between neural modelling and experimental neuroscience, 1997.

- [377] T. Wennekers and G. Palm. Syntactic sequencing in hebbian cell assemblies. *Cogn Neurodyn*, 3(4):429–441, Dec 2009.
- [378] F. Werner, F. Maire, J. Sitte, H. Choset, S. Tully, and G. Kantor. Topological SLAM Using Neighbourhood Information of Places. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'09*, pages 4937–4942, Piscataway, NJ, USA, 2009. IEEE Press.
- [379] J. Widloski and I. R. Fiete. A model of grid cell development through spatial exploration and spike time-dependent plasticity. *Neuron*, 83(2):481–495, Jul 2014.
- [380] A. M. Wikenheiser and A. D. Redish. Hippocampal theta sequences reflect current goals. *Nat Neurosci*, 18(2):289–294, Feb 2015. Article.
- [381] T. J. Wills, C. Barry, and F. Cacucci. The abrupt development of adult-like grid cell firing in the medial entorhinal cortex. *Front Neural Circuits*, 6:21, 2012.
- [382] T. J. Wills, F. Cacucci, N. Burgess, and J. O’Keefe. Development of the hippocampal cognitive map in preweanling rats. *Science*, 328(5985):1573–1576, Jun 2010.
- [383] T. J. Wills, L. Muessig, and F. Cacucci. The development of spatial behaviour and the hippocampal neural representation of space. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1635):20130409–20130409, dec 2013.
- [384] S. S. Winter, B. J. Clark, and J. S. Taube. Disruption of the head direction cell network impairs the parahippocampal grid cell signal. *Science*, 347(6224):870–874, 2015.
- [385] A. P. Witkin. Scale-space Filtering. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence - Volume 2, IJCAI'83*, pages 1019–1022, San Francisco, CA, USA, 1983. Morgan Kaufmann Publishers Inc.
- [386] M. P. Witter. Intrinsic and extrinsic wiring of CA3: indications for connectional heterogeneity. *Learn. Mem.*, 14(11):705–713, Nov 2007.
- [387] R. O. L. Wong and A. Ghosh. Activity-dependent regulation of dendritic growth and patterning. *Nat Rev Neurosci*, 3(10):803–812, Oct 2002.
- [388] M. M. Yartsev, M. P. Witter, and N. Ulanovsky. Grid cells without theta oscillations in the entorhinal cortex of bats. *Nature*, 479(7371):103–107, Nov 2011.
- [389] A. J. Yepes, J. Tang, S. Saxena, T. Brosch, and A. Amir. Weighted population code for low power neuromorphic image classification. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 4294–4301, July 2016.

- [390] D. Yoganarasimha. Head direction cell representations maintain internal coherence during conflicting proximal and distal cue rotations: Comparison with hippocampal place cells. *Journal of Neuroscience*, 26(2):622–631, Jan 2006.
- [391] K. Yoon, M. A. Buice, C. Barry, R. Hayman, N. Burgess, and I. R. Fiete. Specific evidence of low-dimensional continuous attractor dynamics in grid cells. *Nat Neurosci*, 16(8):1077–1084, Aug 2013. Article.
- [392] R. A. Young. The Gaussian derivative model for spatial vision: I. Retinal mechanisms. *Spat Vis*, 2(4):273–293, 1987.
- [393] F. Zenke, G. Hennequin, and W. Gerstner. Synaptic plasticity in neural networks needs homeostasis with a fast rate detector. *PLOS Computational Biology*, 9(11):1–14, 11 2013.
- [394] S. Zhang, F. Schonfeld, L. Wiskott, and D. Manahan-Vaughan. Spatial representations of place cells in darkness are supported by path integration and border information. *Front Behav Neurosci*, 8:222, 2014.
- [395] S. Zhang and D. Manahan-Vaughan. Spatial olfactory learning contributes to place field formation in the hippocampus. *Cerebral Cortex*, 25(2):423, 2015.
- [396] B. Y. Zhao, L. Huang, J. Stribling, S. C. Rhea, A. D. Joseph, and J. D. Kubiatowicz. Tapestry: A resilient global-scale overlay for service deployment. *IEEE J.Sel. A. Commun.*, 22(1):41–53, September 2006.
- [397] R. Zhong, G. Li, K. L. Tan, L. Zhou, and Z. Gong. G-tree: An efficient and scalable index for spatial search on road networks. *IEEE Transactions on Knowledge and Data Engineering*, 27(8):2175–2189, Aug 2015.
- [398] E. Zilli. Models of grid cell spatial firing published 2005-2011. *Frontiers in Neural Circuits*, 6:16, 2012.
- [399] F. Zuber and R. Douglas. A framework for modeling the growth and development of neurons and networks. *Front Comput Neurosci*, 3:25, 2009.
- [400] M. B. Zugaro, A. Arleo, A. Berthoz, and S. I. Wiener. Rapid spatial reorientation and head direction cells. *J. Neurosci.*, 23(8):3478–3482, Apr 2003.