Learning long temporal sequences in spiking networks by multiplexing neural oscillations

Philippe Vincent-Lamarre

A thesis submitted to the University of Ottawa
in partial fulfilment of the requirements for
the Degree of Doctor of Philosophy in Experimental Psychology

School of Psychology
University of Ottawa
Ottawa, Canada

SUMMARY

Many living organisms have the ability to execute complex behaviors and cognitive processes that are reliable. In many cases, such tasks are generated in the absence of an ongoing external input that could drive the activity on their underlying neural populations. For instance, writing the word "time" requires a precise sequence of muscle contraction in the hand and wrist. There has to be some patterns of activity in the areas of the brain responsible for this behaviour that are endogenously generated every time an individual performs this action. Whereas the question of how such neural code is transformed in the target motor sequence is a question of its own, their origin is perhaps even more puzzling. Most models of cortical and sub-cortical circuits suggest that many of their neural populations are chaotic. This means that very small amounts of noise, such as an additional action potential in a neuron of a network, can lead to completely different patterns of activity. Reservoir computing is one of the first frameworks that provided an efficient solution for biologically relevant neural networks to learn complex temporal tasks in the presence of chaos. We showed that although reservoirs (i.e. recurrent neural networks) are robust to noise, they are extremely sensitive to some forms of structural perturbations, such as removing one neuron out of thousands. We proposed an alternative to these models, where the source of autonomous activity is no longer originating from the reservoir, but from a set of oscillating networks projecting to the reservoir. In our simulations, we show that this solution produce rich patterns of activity and lead to networks that are both resistant to noise and structural perturbations. The model can learn a wide variety of tem-

poral tasks such as interval timing, motor control, speech production and spatial navigation.

REMERCIEMENTS

Je tiens à remercier mon directeur de thèse, Dr Jean-Philippe Thivierge, pour tout le soutien, l'encadrement et son implication dans le travail que j'ai fait durant ce doctorat.

Merci à Dr Jean-Claude Béïque de m'avoir donné l'opportunité de développer mes compétences en neuroscience expérimentale, et d'avoir cru en moi pour laisser entreprendre des projets de recherche dans son laboratoire. Un merci spécial à Dr Stevan Harnad qui a été une source d'influence importante sur ma façon d'aborder la recherche scientifique.

J'aimerais aussi remercier tous mes collègues, pour leur présence et les discussions intéressantes que j'ai eues avec eux, notamment (mais pas exclusivement!): Dr Eric S. Kuebler, Nareg Berberian, Matias Calderini, Éloïse Giraud, Matias Calderini, Michael Lynn, Jean-François Boucher, Emerson Harkin et Sébastien Maillé.

J'aimerais remercier les membres de mon comité de thèse, Dr Sylvain Chartier, Dr Denis Cousineau, Dr Richard Naud ainsi que Dr Brent Doiron.

Puis je tiens à remercier mes parents, Gisèle Lamarre et Michel Vincent, ma soeur Sarah-Jane Vincent-Lamarre et toute ma famille et mes amis pour leur soutien, sans lequel je n'aurais certainement pas eu l'énergie et la résilience nécessaire pour effectuer ce travail.

Finalement, un merci spécial à ma femme, Jade Boivin, qui a toujours cru en moi, et qui a vécu avec moi tous les hauts et les bas traversés depuis le début de mon parcours académique.

Contents

1 Chapter 1 1

1.1 Temporal processing . . . 3

1.2 Recurrent neural networks . . . 12

1.3 Connectionist models . . . 12

1.4 Reservoir computing . . . 17

1.5 Neural dynamics of temporal processing . . . 22

1.6 Outline of this work . . . 33

2 Chapter 2 35

2.1 Introduction . . . 36

2.2 Methods . . . 39

2.3 Results . . . 49

2.4 Discussion . . . 75

3 Chapter 3 83

3.1 Introduction . . . 84

3.2 Methods . . . 86

3.3 Results . . . 94

3.4 Discussion . . . 115

4 Chapter 4 121

4.1 Structural perturbation in chaotic systems . . . 122

4.2 Neural oscillations . . . 123

4.3 Concluding statement . . . 127

5 Appendix 129

List of Figures

Figure 1 Schematic representation of network architectures. 41

Figure 2 Activity of intact and damaged networks. 50

Figure 3 Performance of innate training on the timing task after perturbations. 54

Figure 4 Performance on the timing task after different types of perturbations. 55

Figure 5 Distribution of errors after a global perturbation with standard FORCE. 56

Figure 6 Clamping neurons from reservoirs of different sizes. 57

Figure 7 Eigenvalues of perturbed networks. 58

Figure 8 Clamping neurons from reservoirs of different sizes. 62

Figure 9 Error distribution of networks with an exponential, power-law and modular connectivity. 65

Figure 10 Generation of stable trajectories with oscillators. 67

Figure 11 Performance of the driven model. 68

Figure 12 Saturation and correlation (see Methods) of chaotic networks with different gain values. 73

Figure 13 Impact of the network parameters on its dynamics. 74

Figure 14 Perturbation on the three bit flip-flop task. . . . 77

Figure 15 Retraining networks after perturbations. 78

Figure 16 Oscillation driven spiking recurrent network as a reservoir computing model. 97

Figure 17 Parameter exploration for training a reservoir network. 98
Figure 18 Reservoir driven with intrinsically generated oscillations. 99
Figure 19 Autonomous production of repeatable periodic activity. 100
Figure 20 Parameter exploration of the network's synchronous activity (inhibition conductance and kinetics). . . 101
Figure 21 Parameter exploration of synchronous activity (excitatory conductance and p). 102
Figure 22 Impact of reservoir feedback to the oscillatory networks. 103
Figure 23 Parallel training of multiple tasks. 104
Figure 24 Temporal rescaling of the network's activity. . . 107
Figure 25 Network performance with desynchronized inputs. 108
Figure 26 Network dynamics and performance with temporal rescaling. 109
Figure 27 Speech learning and production with temporal rescaling. 110
Figure 28 Formation of place cell sequences and replay. . . 114
Figure 29 Dynamics underlying trial-averaged temporal tuning. 132
Figure 30 Modelling dynamics of trial-averaged peaks. . . 135
Figure 31 Detection of peak activity with trial-averaged data. 136

List of Tables

Table 1 Temporal processing with the main models in computational neuroscience. 23

Table 2 Parameters used for the different architectures. . 44

Table 3 Reservoir parameters. 88

[1] *School of Psychology and Center for Neural Dynamics, University of Ottawa, Ottawa, Ontario, Canada*

PUBLISHED WORK

This thesis document is based on the following published work: Vincent-Lamarre, P., Lajoie, G., & Thivierge, J.-P. (2016). Driving reservoir models with oscillations: A solution to the extreme structural sensitivity of chaotic networks. *Journal of Computational Neuroscience*, 41(3), 305–322.

As well as the following submitted work: Vincent-Lamarre, P., Calderini, M. & Thivierge, J.-P. Learning long temporal sequences in spiking networks by multiplexing neural oscillations.

ABBREVIATIONS

5-HT: serotonin

ADALINE : adaptative linear neuron

ANN: artificial neural network

BAM: bidirectional associative memory

CAP: credit assignment problem

CFC: cross-frequency coupling

CST: cortico-striato-thalamic

DA: dopamine

DLPFC: dorsolateral prefrontal cortex

fMRI: functional magnetic resonance imaging

FORCE: first-order reduced and controlled error

LFP: local field potentials

LSM: liquid-state machines

LSTM: long short-term memory networks

LTD: long-term depression

LTP: long-term potentiation

ms: millisecond

MLP: multi-layer perceptron

MPO: membrane potential oscillations

NE: norepinephrine

PA: pace-maker accumulator

RC: reservoir computing

ReLU: rectified linear units

RLS: recursive least-square

RPE: reward prediction error

RNN: recurrent neural network

SDC: state-dependent computing

SET: scalar expectancy theory

VGP: vanishing gradient problem

VTA: ventral tegmental area

1 CHAPTER 1: GENERAL INTRODUCTION

Temporal processing is an essential component of the behaviour and cognitive capacities of biological organisms. For instance, the simplest type of behaviour, like moving a limb, requires the nervous system to contract muscles or move some joints in a predetermined sequence. Then, from those more primitive forms of temporal processing there is a wide range of processes that various organisms are able to exhibit, like interval timing, speech generation and processing, throwing a ball or playing music. Even though temporal processing is critical to sensorimotor processing and learning, we still know very little about the neural circuits that encode temporal information (Paton & Buonomano, 2018). Mauk & Buonomano (2004) classified temporal processes in four main categories: microsecond processing, millisecond processing, second processing and circadian rhythms. Microsecond processing is involved in sound localization and echolocation, millisecond timing includes behaviours such as speech and motor control, second processing includes foraging, decision-making and conscious time estimation, and circadian timing includes appetite control and the sleep-wake cycle. Different brain structures and processes have been suggested to play important roles for each of these behaviours. Circadian timing is perhaps the best understood of all scales. We know that the suprachiasmatic nucleus of the hypothalamus is the seat of the clock regulating metabolic and behavioural rhythms (Reppert & Weaver, 2002). The underlying mechanism for this type of timing relies on a network of molecular of transcriptional feedback loops (Darlington et al., 1998). The other end of the spectrum, microsecond processing, is also relatively well understood. For instance, a seminal finding on sound localization showed that its core

mechanism relies on the detection of an interaural time differences by the brain stem. The time of arrival of a sound to both ears can be used to identify the angle of the sound relative to the head. The conduction velocity of the axons originating from the cochlear nucleus (C. Carr & Konishi, 1990) creates a constant delay that is integrated by the neurons of the nucleus laminaris to identify to source of a sound. Similarly, part of the visual motion detection originates from the retina, where the sequential activation of ON and OFF ganglion cells sends a readily accessible representation of motion to the visual cortex (Rodieck, 1965). As opposed to computations occurring in very fast or very slow timescales, the other two levels are not as well understood. However, it is very likely that the intrinsic properties of individual neurons and networks are critical to shaping the core neural dynamics involved at this time-scale (Paton & Buonomano, 2018). In the first chapter of this work, we used a framework termed reservoir computing (RC) to model temporal tasks in the millisecond to second range. We first show the limitations of this model, which is extremely sensitive to structural perturbations (e.g. clamping a single neuron to resting potential for the duration of a simulation). We show that driving the model with oscillations dramatically increase the resilience of the network to such perturbations, while conferring some beneficial computationally and biologically relevant properties to the model. We expand upon this modified model in the second chapter, and we transition from rate-units reservoirs to a realistic biologically plausible network of spiking units. We show that mixing oscillations with different periods in a recurrent neural network (RNN) offers a highly powerful mechanism to control its activity, even when operating in a chaotic regime. In the second chapter, we expand onto this finding, and use a more biologically relevant RNN of spiking

units with realistic biophysical properties. We show that this model can perform tasks such as motor control and speech production, and can capture the neural dynamics of place cells sequences in the hippocampus. Additionally, we show that this new model can perform temporal rescaling in an intuitive and efficient manner.

1.1 Temporal processing

It is puzzling how neuronal circuits can produce the type of activity necessary to generate the robust and flexible dynamics necessary for the survival of most living organisms. Conrad Waddington stated that "the main respect in which the biological picture is more complicated than the physical one is the way time is involved in it" (Waddington, 2014). In respect to neural networks, this challenge is highlighted by the fact that many cortical and subcortical circuits appear to be highly chaotic (C. van Vreeswijk & Sompolinsky, 1996; C. v. van Vreeswijk & Sompolinsky, 1998), which means that infinitesimal perturbations of their state can lead to exponential divergence in their activity with time. In other words, in various conditions associated with biological neural networks, the amount of noise that they contain is an exponential function of time. This means that even if the state of a large population of neurons is reset to the exact same configuration upon beginning a task unfolding in time, the addition of a single spike can lead to drastically different patterns of activity. This contrast with the apparent robustness of temporal behaviours that can be performed by most living organisms (Buonomano & Maass, 2009). A trivial workaround to this issue would be to provide external stability to neural networks (Abbott, DePasquale, & Memmesheimer, 2016). Intuitively, if the same time-varying current is repeatedly injected in a

network of neurons (even chaotic Molgedey, Schuchhardt, & Schuster, 1992), one might expect some regularity in the network's activity across trials. This strategy is probably used by brain regions involved in sensory processing, where the underlying neural circuits are usually driven by inputs external to the organism (Yamins & DiCarlo, 2016), which provides some sort of regularity and structure to the brain's dynamics. For instance, streams of photons shape the retinal activity and sound waves drive the activity of cochlear hairy cells (Kandel et al., 2000). Such environmental drive provides a robust and repeatable input (e.g. seeing an apple) which results in the reliable and structured neural activity that is required to recognize a stimulus (e.g. recognizing it as an apple). This is an oversimplification, because there is some inherent noise in both the inputs (e.g. different angles, luminosity when viewing an apple), and the neural networks processing it that has to be dealt with. But there is at least enough regularity in those stimuli, which allows the brain to pick up on the invariant sensorimotor features required to do the right thing with the right kind of thing (Harnad, 2005). However, most temporal behaviours (e.g. reaching motion, playing piano, interval timing or speech) occur without such external inputs (Abbott et al., 2016; Buonomano & Maass, 2009). Structured time-varying inputs to chaotic networks underlying autonomous tasks in the absence of external drive would have to be generated elsewhere in the brain. We then run into a problem of circularity, because the autonomous production of structured time-varying patterns of activity is what we are trying to explain in the first place. So how are networks of neurons that are noisy and chaotic able to provide the structured activity required to execute the tasks that don't have an external drive alongside them? Is there any way to break out of the circularity where a reliable and repeatable

time-varying input is required to generate a reliable and repeatable time-varying output? Before attempting to provide a solution to this problem, we will discuss the behaviours and cognitive processes that it applies to. We will then discuss what we know about the neural dynamics associated with those tasks, and then provide an overview of the different approaches used to model those tasks.

1.1.1 *Behavioural and cognitive tasks*

There are no tasks that are purely timeless, given that the universe evolves both in space and time. Therefore, even the tasks that seem discrete in time involve a continuous stream of information with temporal dependencies. However, we can formalize tasks as being either discrete (with a recognizable beginning and end) and continuous (no recognizable beginning and end) in time (R. A. Schmidt, Lee, Winstein, Wulf, & Zelaznik, 2018). The type of input (stimulus) on the other hand can be classified as static (not changing in time) or dynamic (changing in time) (Gupta, Jin, & Homma, 2004). For instance, at one end of the spectrum, recognizing a scene on a picture (e.g. a cat falling of a desk) is a discrete process (you recognize the scene or not) that relies on static information (the colors shades and intensity of the picture are not changing in time). Then, if you observe the scene in real life, the recognition task is still discrete, but the information is now dynamic (you can see the motion of the cat falling towards the floor). Then you might want to run towards the cat and reach out both of your hands (continuous task, dynamic input) to prevent a tragic collision with the floor. Performing the same action when exposed to the picture (continuous task, static input) is probably not as useful (unless you want to grab it for your collection of falling cat pictures). Regardless of the type of input involved, all types of tem-

poral processes share some common features. According to Mauk & Buonomano (2004), static and dynamic information involves distinct neuronal processes. They refer to these different processes as spatial and temporal. Spatial processing relies on a snapshot of which neurons are activated by a sensory input, whereas temporal processing requires to take into account the temporal pattern of action potentials generated by the input. These processes are not independent, and many cognitive and behavioural tasks rely on a spatio-temporal code (Buonomano & Maass, 2009). As stated earlier, the timescales that are likely to rely on computations at the level of networks of neurons are the millisecond to seconds timescales. Millisecond timing includes behaviours such as motor control (Churchland et al., 2012), playing and recognizing music (Levitin, 2009; Tchernichovski, Mitra, Lints, & Nottebohm, 2001), speech generation (Engineer et al., 2008) and recognition (Hickok & Poeppel, 2007). Second timing involves behaviours such as conscious time estimation (Grondin, 2010), generation and perception of sequences of events (Hardy, Goudar, Romero-Sosa, & Buonomano, 2018) and timed conditioned responses (Rakitin et al., 1998). There is not a clear boundary between the two timescales. For instance, generating a motor sequence can last a few milliseconds (reaching for an object) up to minutes (such as generating a Morse code sequence or playing a song on the piano). In this work, we are mostly interested in the millisecond to seconds timescales. More specifically, we are interested with continuous tasks, where a static or dynamic input (with a source external or internal to the organism) to a neural population is converted into a continuous output. We will attempt to explain how cortical circuits can generate the activity necessary to approximate a timed interval of a few seconds long, where with the exception of a "GO" cue, no external input is available to

drive the activity of a neural population. Thus, our aim is to understand the neural systems involved in "autonomous" temporal processing, where the duration of the task has to largely exceed the duration of the input. However, some overlap exists between the mechanisms underlying different types of temporal processing, so our discussion will also cover other types of task.

1.1.2 *Models of temporal processing*

There are many models that have been suggested to explain autonomous temporal processing. These models can be divided in two main subtypes: dedicated and intrinsic models (Ivry & Schlerf, 2008). Dedicated models of temporal processing are modular, and assume that the unique neural structure of some brain regions allow them to process temporal information. Such model assumes that a dedicated model has a "unique representational capability and is accessed whenever a particular task requires precise timing" (Ivry & Schlerf, 2008, p.274). Different brain regions have been hypothesized to implement dedicated interval timing mechanisms, such as the basal ganglia (Harrington, Haaland, & Hermanowitz, 1998), the prefrontal cortex (Lewis & Miall, 2006) and the motor cortex (Macar, Coull, & Vidal, 2006). The main argument in favor of dedicated systems comes from lesion studies, where temporal processing can be selectively disrupted for a given modality (e.g. audition Harrington & Haaland, 1999), while the other aspects of sensory processing for this modality appear to be preserved. Therefore, if only temporal processing appears to be impaired when a neural population is removed, it might be because its only purpose was to process temporal information. On the other hand, intrinsic models assume that temporal information is inherent to the neural dynamics of some brain systems, and arise from the gen-

eral properties of populations of interconnected neurons (Karmarkar & Buonomano, 2007). Therefore, temporal information can be processed by the internal dynamics of many neural populations which are not necessarily dedicated to processing this type of information (Burr, Tozzi, & Morrone, 2007). The main takeaway from this view is that some brain networks are multi-purpose processing units that can represent both temporal information and process modality specific information (Burr et al., 2007). Although still supported by some experimental findings, the dedicated models were more popular in the earlier days of the research on temporal processing. More recent models (including the model we are using in this work), are closer to the intrinsic view and are supported by findings of modern neuroscience. We will now give a brief overview of the main models of temporal processing.

1.1.3 *Sequential models*

One of the first classes of models that were introduced is the delay line model (C. E. Carr & Konishi, 1988; Jeffress, 1948; Licklider, 1951). This dedicated model suggests that because of the axonal conduction velocity, chains of neurons can be connected to form a delay line that determines the timing of an action potential propagating to downstream areas. Specialized circuits of neurons could then monitor the different delays to decode timing related information. Although appropriate for modelling tasks that require very short delays like for the interaural time difference for sound source localization (C. Carr & Konishi, 1990), this mechanism does not operate on a timescale plausible for longer tasks. A similar idea was used by (Abeles, 1982) to describe synfire chains. Instead of using sequences of connections between individual neurons, synfire chains stipulates that pools of

neurons are connected in a feed-forward manner. This conception of neural connectivity has the benefit of being more general and is therefore also consistent with the intrinsic models. That is because such patterns of connectivity can be embedded in populations of recurrent units (Trengove, van Leeuwen, & Diesmann, 2013; Vincent, Tauskela, & Thivierge, 2012; Weissenberger, Meier, Lengler, Einarsson, & Steger, 2017). Synfire chains have gained in vivo and in vitro support recently (Long, Jin, & Fee, 2010) in the context of song generation in the HVC neurons of zebra finches (Long & Fee, 2008). However, synfire chains suffer from the same limitations as delay lines, which is that the duration of the temporal information they can represent is limited to a few hundreds of millisecond at best (Trengove et al., 2013).

1.1.4 *Clock models*

Another popular class of models are pacemaker-accumulator (PA) models, also termed clock models. In this class of models, a neural system is divided in three components: 1- a pace-maker (clock) that emits pulses at a constant rate, 2- an accumulator that increases its activity linearly with every pulse, and 3- an attentional gate that resets the activity of the accumulator at the start of the task (Simen, Rivest, Ludvig, Balci, & Killeen, 2013; Treisman, 1963). Originally introduced by (Treisman, 1963) as the PA model, an equally popular framework was introduced later as the scalar expectancy theory (SET) (Gibbon, 1977), which mainly differs by the way it estimates a scalar interval based on the state of the accumulator. In addition to its simplicity and how intuitive it is, this model initially gained a lot of support based on how readily it can account for Weber's law (Treisman, 1963). According to this law, the standard deviation of timing estimations scales linearly with the duration of the estimated interval, which

leads to a constant coefficient of variation (Gibbon, 1977). However, conclusive evidence is still lacking to support that the brain might use either model (Kononowicz & van Wassenhove, 2016). A second class of models that uses clocks are beat frequency models. In this model, a series of oscillators with different periods are activated by a stimulus onset (Miall, 1989). Where the previous model needed an accumulator to keep track of the number of pulses emitted by the pace-maker, this model relies on the coincidental activation of a sub-population of oscillators to detect a given timing interval. The activity generated by the oscillators creates a pattern that is unique for a duration equal to the least common multiple of the periods of all oscillators (Matell & Meck, 2000; Miall, 1989). Some efforts were made to establish the potential neurological underpinning of this model, which lead to the Striatal beat frequency (SBF) theory (Matell & Meck, 2004). This model emphasizes the role of the basal ganglia as a coincidence detector that monitors cortical oscillators of different frequencies that are spread throughout the cortex (Buhusi & Meck, 2005). However, only little evidence supports the model (Matell, 2014). It also suffers from the same limitation as the PA models, as they rely on a coincidence detection mechanism that is only suited for interval timing tasks, and have no application for other complex temporal tasks.

1.1.5 *State-dependent models*

A more recent approach termed state-dependent computing (SDC) (Buonomano & Maass, 2009) assumes that temporal information is encoded in the spatio-temporal activity of networks of neurons. That is, at each time step during a task, there is a different sub-population of neurons that are active, and this sequence of activation conveys the task-relevant information. This approach assumes that the in-

teraction between the structure of the network (e.g. synaptic connections or electrophysiological properties of its neurons), its current state (e.g. membrane potential of each cell or synaptic efficacy), and external stimulations shape the spatio-temporal patterns of activity within the network (Buonomano & Maass, 2009). Surprisingly, this type of model is well suited for all types of tasks, including discrete tasks with static inputs. For instance, studies in the locust suggest that odour recognition in this organism is achieved by discriminating between different neural trajectories evoked by an odour (Broome, Jayaraman, & Laurent, 2006). This finding departs from the traditional view of static attractor networks where the activity of the network converges to a steady state (Amit & Amit, 1992; Hopfield, 1982; Yuste, 2015) as a mechanism for categorization. This finding highlights the complexity of the temporal processing landscape, where even categorizing static inputs can require discrimination between neural activity evolving over time (Brody, Romo, & Kepecs, 2003; Rabinovich, Huerta, & Laurent, 2008). This view of information encoded in the spatio-temporal neural trajectories has many interesting properties for other applications such as interval timing (Buonomano, 2000; Laje & Buonomano, 2013), motor control (Churchland et al., 2012; Sussillo & Abbott, 2009), speech processing or playing music (Nicola & Clopath, 2017). In this work, we investigate models that are based on the SDC framework. We studied one implementation using RNNs, reservoir computing, which we will introduce in the next section, after covering the main types of RNN models since they were first introduced. Then, we will review the brain dynamics associated with temporal processing, and discuss their relevance for this type of information processing.

1.2 Recurrent neural networks

SDC makes several assumptions about the way neural circuits are organized and how their activity can be decoded (Buonomano & Maass, 2009). Generally speaking, this class of model goes hand-in-hand with neural networks that have a recurrent connectivity (Hopfield, 1982; C. van Vreeswijk & Sompolinsky, 1996). This is because in addition to being a common model of brain networks (C. van Vreeswijk & Sompolinsky, 1996; Yuste, 2015), such networks are dynamic systems that can intrinsically represent complex patterns of activity (Sussillo, 2014). From the perspective of a neuroscientist, it seems fairly obvious that models of brain circuits should involve a recurrent connectivity. However, for most of its history, the field of artificial neural networks (ANN) relied on purely feedforward architectures (Schmidhuber, 2015). Part of the reason behind that choice is that most developments in this field have been driven through the field of machine learning, where researchers are mostly motivated by the performance of the models and not their biological relevance.

1.3 Connectionist models

The development of the first ANNs was directly motivated by the brain architecture. McCulloch & Pitts (1943) wanted to understand how distributed units like neurons in the brain could perform basic computations, and they created a model sharing their name (McCulloch-Pitts) to do so. Their model had binary input units projecting to a thresholded output unit, and by changing the polarity of the connections between the two layers, they were able to implement boolean functions such as AND, OR and NOT. However, their model had no

training algorithm, which meant that the weights and thresholds had to be manually selected. A solution to this limitation was provided more than a decade later, when Rosenblatt (1958) introduced the perceptron, which has a similar feedforward architecture. Inspired by the work of Hebb (2005), Rosenblatt developed an algorithm able to tune the thresholds and connections of the neurons so they could "learn" a task. Perceptrons were able to learn how to classify an input, and another model named adaptative linear neuron (ADALINE) could perform a regression that gave a continuous output (Widrow & Hoff, 1962). Despite initial enthusiasm, the hype surrounding the potential of these networks was eventually dampened by the seminal work of Minsky & Papert (1969), which showed that networks with only two layers (input and output) such as perceptrons and ADALINE could only learn linearly separable functions, and could not learn non-linear functions such as XOR.

1.3.1 *Multi-layer networks*

In the same work, Minsky & Papert (1969) identified that such tasks could be solved by multiple layers of perceptrons, but showed that the learning algorithm of (Rosenblatt, 1958) couldn't learn those functions. The main bottleneck was the credit assignment problem (CAP) (Minsky, 1961), which highlights the difficulties in assigning the responsibility over the parameters of a network for the output error. And this problem gets increasingly difficult to solve as the depth of a network increases. It took the popularization of the backpropagation algorithm by Rumelhart, Hinton, & Williams (1988) to train multilayer perceptrons (MLP), and overcome the CAP in order to revive the interest in neural networks. This learning rule opened a wide range of possible tasks to be learned by MLPs (Duda, Hart, & Stork, 2012),

and is still used by the state-of-the-art deep neural networks (Goodfellow, Bengio, & Courville, 2016; LeCun, Bengio, & Hinton, 2015). The core concept behind this algorithm is that the network's weights are modified proportionally to the partial derivative (gradient) of the error function with respect to the current weight (Rumelhart et al., 1988). The algorithm got its name from the fact that it sends the error backwards across the network's layer, providing a partial error signal that can be credited to the hidden neurons involved in the earlier steps of input processing. This feature of the learning algorithm has a fundamental flaw that prevented it from having a similar impact on the efficiency of RNNs: the vanishing gradient problem (VGP) (Bengio, Simard, & Frasconi, 1994; Hochreiter, Bengio, Frasconi, & Schmidhuber, 2001). The problem is that the error gradient decreases exponentially when propagated across many layers (typically when the connection weights are smaller than 1), and becomes vanishingly small which doesn't allow the network to change its weights in order to improve its performance. On the converse, modifications to some network parameters such as changing the activation functions from sigmoids to rectified linear units (ReLU) (Hahnloser, Sarpeshkar, Mahowald, Douglas, & Seung, 2000), or initializing the network with weights greater than 1 could lead to exploding gradients (Pascanu, Mikolov, & Bengio, 2012). This strongly impeded the development of RNNs, which are in essence deep feedforward networks unfolded in time (Harrington et al., 1998). The same limitations severely hampered the field of deep learning and lead to the artificial intelligence winter (Hendler, 2008). However, the solutions that launched back the popularity of deep learning couldn't be applied as easily to RNNs (Goodfellow et al., 2016). However, simple motifs embedded in the connectivity of shallow feed-forward networks can still process tem-

poral activity on a relatively long time-scale (Alluri et al., 2016; Naud, Houtman, Rose, & Longtin, 2015).

1.3.2 *Associative memory networks*

There were still noteworthy developments with RNNs when most of the attention was directed toward MLPs in the 80s. The first generation of RNNs are associative memory networks, with the Hopfield network (Hopfield, 1982) as the first implementation, and subsequently bidirectional associative memory (BAM)(Chartier & Boukadoum, 2006; Kosko, 1988). Hopfield networks were powerful models that kick-started the research on attractor networks (Amit & Amit, 1992). However, this family of RNNs are typically used in tasks where they converge to point-attractors (Milnor, 1985) to associate static patterns of input-output, although they can learn more complex temporal dynamics Chartier & Boukadoum (2011).

1.3.3 *Modern RNNs*

Many types of RNNs architecture and learning rules have been developed in the last three decades (Salehinejad, Sankar, Barfett, Colak, & Valaee, 2017), but there are two main approaches that lead to noteworthy achievements. The state-of-the-art approaches in terms of performance and for their range of applications are the long short-term memory networks (LSTM) (Hochreiter & Schmidhuber, 1997). Although officially in the family of deep networks, this architecture is the first type of RNN that was able to offer a cutting-edge performance in many applications (Goodfellow et al., 2016; LeCun et al., 2015). In addition to being a powerful solution to many tasks involving static inputs, one of its main features is that it is able to process continuous information. This means that the network is able

to process the temporal dependencies between sequences of discrete inputs. As their name suggests, LSTMs possess a long-term memory provided by their recurrent connections which are modified based on long sequences of input. The short term component refers to impact of recent inputs on the hidden state of the network. Put simply, the model is able to deal with the exploding and vanishing gradients by keeping their connection weights near 1, and therefore keep the value of the gradient constant on average, as it back-propagates (Hochreiter & Schmidhuber, 1997). LSTMs are one of the most popular choices today (Sainath, Vinyals, Senior, & Sak, 2015), but they remain of limited interest for neuroscientists because they neglect many aspects of biological neural networks (Kietzmann, McClure, & Kriegeskorte, 2019). Although deep learning offers a useful tool for the analysis of neural data (Pandarinath et al., 2018), they offer little insight as models of neuronal dynamics. Most implementations use continuous rate-units whereas the neurons in the brain are spiking which provides a temporal code of binary states (Diesmann, Gewaltig, & Aertsen, 1999; Gerstner, Kempter, van Hemmen, & Wagner, 1996; Mainen & Sejnowski, 1995; Tiesinga, Fellous, & Sejnowski, 2008). Additionally, the back-propagation algorithm has severe limitations in terms of biological plausibility because it requires some non-local transport of synaptic weights (Bengio, Lee, Bornschein, Mesnard, & Lin, 2015). This means that earlier layers of deep networks need to gain access to the information in downstream stages of the circuit (Bengio et al., 2015). Recent developments have however suggested alternatives to backpropagation for the CAP by using feedback connections towards earlier layers (Lillicrap, Cownden, Tweed, & Akerman, 2016), and by exploiting the non-linearity of dendritic integration with top-down feedback (Guerguiev, Lillicrap, & Richards, 2017; Richards & Lillicrap, 2019). Efforts

have also been made to implement more biologically relevant features in deep learning, such as spiking neurons, but these models remain underperforming (Hunsberger & Eliasmith, 2016; Tavanaei & Maida, 2016). It is also unclear to what extent they can advance the field of neuroscience given that the interests of the machine learning community towards these models are more about their performance, mostly because of their high energy efficiency (Verzi et al., 2018), and not as models of the brain.

1.4 Reservoir computing

RC is the second approach to RNNs that achieved significant performance on a broad range of applications. Fundamentally, it is a fairly simple type of model compared to many deep learning methods. RC is an overarching term that includes many sub-types of models without clear boundaries between them, such as liquid state-machines, echo-state networks and state-dependent networks (Buonomano & Maass, 2009). The approach was introduced independently by two groups in the early 2000s. The framework refers to models that typically have the following features (Lukoševičius & Jaeger, 2009):

1. An input layer.
2. A reservoir: a hidden randomly connected recurrent layer with non-linear activation functions, static weights and sparse connectivity.
3. A linear readout unit with tunable connections (the recursive least-square (RLS) Haykin, 2002 being the most popular learning algorithm) .

Reservoirs get their name from the fact that they behave like a pool of water. If a pebble is thrown in liquid, its impact will cause ripples that will last for some amount of time, which creates a sort of memory of past events. The first of the two approaches, echo-state networks (Jaeger, 2002) have reservoirs of rate-units, receives continuous streams of inputs, are mostly used for discrete tasks of discrimination and converge towards a stable attractor when not driven by an external input. That is because the model performs better at the edge of chaos (Bertschinger & Natschläger, 2004), where the spectral radius of the network is lower than 1 (Zhiping, 2009), which leads to stability in the absence of input (Jaeger, 2002). This approach has provided state-of-the-art performances in some applications such as stock market predictions (Lin, Yang, & Song, 2008), wireless communication (Jaeger & Haas, 2004), robot motor control (Ishii, van der Zant, Becanovic, & Ploger, 2004) and speech recognition (Skowronski & Harris, 2007). This approach is typically related to the field of machine learning and isn't designed as a tool to model neural dynamics. The second approach in RC are liquid-state machines (LSM) (Maass, Natschläger, & Markram, 2002). It shares similar properties to ESNs, but use biologically inspired parameters, such as reservoirs of spiking neurons instead of rate-units, and is therefore a more neuroscience oriented model. LSMs quickly gained attention as a possible model for cortical circuits (Joshi & Maass, 2005), as they were introduced as an abstract cortical microcolumn model (Maass, Natschläger, & Markram, 2004). The two approaches exploit the same set of basic features provided by a reservoir. First, reservoirs create a high-dimensional representation of the incoming stream of inputs, which facilitate their separation (Häusler, Markram, & Maass, 2003). This allows the readouts to use local rules to train their connections. In

addition to being more compatible with biological constraints, this reduces the complexity of the learning rule optimization (Schrauwen, Verstraeten, & Van Campenhout, 2007). Secondly, the reservoirs are made to operate near the edge of chaos (Bertschinger & Natschläger, 2004; Legenstein & Maass, 2007) by selecting their spectral radius (that is dependent of the strength of the recurrent connections between the reservoir's units Zhiping, 2009). Therefore, having a reservoir that is between stability and chaos will provide the optimal separation of inputs (Bertschinger & Natschläger, 2004). Thirdly, the reservoir units use non-linear activation functions, which is necessary because readout units are typically linear. Therefore, non-linearity in the reservoir allows non-linearly separable inputs to become linearly separable (Lukoševičius & Jaeger, 2009). Lastly, reservoirs possess a fading memory, where the current state of the network is influenced by the past inputs and states (Jaeger, 2002; Maass et al., 2002). Another interesting feature of reservoirs is that because they do not require a predefined connectivity (networks are predominantly random), they fit in the category of intrinsic temporal processing models (Ivry & Schlerf, 2008). Therefore in this paradigm, cortical circuits are general purpose (Maass et al., 2004) temporal processing systems that can be involved in many potential processes simultaneously. For the first few years following their introduction, RC models focused on many types of tasks using both dynamic and static inputs. For instance, discrete tasks with discrete inputs include the boolean operations (and, or, not and Xor) or the "flip-flop" tasks, and requires the network to switch its state depending on brief stimulations from different inputs (Jarne & Laje, 2019). Some tasks involved predicting (continuous task) dynamic non-linear time-series (Jaeger & Haas, 2004). Other tasks were using dynamic inputs such as time-varying input currents to

drive networks in the context of continuous tasks such as motor control (Joshi & Maass, 2005). And other tasks involved classifying (discrete task) spoken digits (dynamic inputs) (Verstraeten, Schrauwen, & Stroobandt, 2006). However, it took some time until some sort of continuous tasks involving static inputs (like interval timing) was successfully modelled by a RC network.

1.4.1 *Chaotic dynamics*

Therefore at the beginning of RC, continuous tasks with static inputs such as interval timing or autonomous motor control were mostly absent. This is mainly because optimal reservoirs that are operating at the edge of chaos are quiescent when not driven by external inputs (Kadmon & Sompolinsky, 2015). Spontaneously active models, on the other hand, are typically chaotic (Brunel, 2000; Sompolinsky, Crisanti, & Sommers, 1988; C. van Vreeswijk & Sompolinsky, 1996). As chaotic systems are exponentially sensitive to small perturbations, it is really hard to train them (Abarbanel, Creveling, & Jeanne, 2008). Surprisingly, one of the first solution to this problem was provided by Jaeger (2001) in a tutorial on RNNs. He used a simple chaotic ESN, and showed that injecting the target function (via feedback connections from the readout) in the reservoir can dampen the chaos and allow the network to generate periodic functions such as sine waves. This idea of using feedback was explored by (Maass, Joshi, & Sontag, 2007), both as a way to keep the beneficial computational properties of the chaotic regime without the disadvantage of the noise, and as a way to extend the capacities of the fading memory of reservoirs. However, the target function had to be pumped in the network during training, which severely limited the biological relevance of this approach.

1.4.2 *FORCE learning*

Sussillo & Abbott (2009) designed the first RC model that was able to capture complex behavioural and neuronal dynamics. They introduced a variation of the feedback driven reservoirs (Maass et al., 2007), and they were able to achieve optimal performance without injecting the target function directly in the reservoir. Instead, they were able to train the readout units in such a way that the output rapidly converged to the target, and termed this method "first-order reduced and controlled error" (FORCE) learning. This algorithm uses strong synaptic modifications which results in low error during all the training phase. Their networks were able to autonomously produce a range of continuous tasks, such as generating periodic outputs, non-linear functions such as a Lorenz attractor and use multiple readouts to reproduce the kinetics of human joints generating a walking and a running motion. This version of RC opened up many possibilities for cognitive modelling with RNNs. Laje & Buonomano (2013) showed that reservoirs could be trained with FORCE learning in order to stabilize innate patterns of activity. This approach allowed the network to learn non-periodic functions, and provided one of the first implementation of RNNs able to perform interval timing tasks in an autonomous manner. That is, with a spontaneously active reservoir, they were able to produce stable and repeatable patterns of activity when a "GO" cue was injected in the network. Therefore, the network was able to execute a continuous task with a transient (50 ms in Laje & Buonomano, 2013) static input. Other implementations involving FORCE learning includes brain-machine interface decoders (Sussillo & Abbott, 2012), production of muscle activity (Sussillo, Churchland, Kaufman, & Shenoy, 2015), producing sequences of activity in RNNs

(Rajan, Harvey, & Tank, 2016) and working memory (Barak, Sussillo, Romo, Tsodyks, & Abbott, 2013; Enel, 2014). FORCE was also adapted to spiking RNNs (Nicola & Clopath, 2017), which allowed the production of a range of tasks such as generating sinusoids, and learning more complex sequences such as playing music, reproducing the mating song of songbirds and learning a short segment of a movie. In this approach, FORCE learning was used to learn non-periodic tasks by implementing them as very long oscillations (the period exceeds the length of the task).

1.4.3 *RNNs as dynamic systems*

In parallel to the endeavour of making RNNs learn useful tasks to model behavioural and cognitive processes, a lot of attention was devoted to study them as dynamic systems. Dynamical systems "consist of a set of variables that describe its state and a law that describes the evolution of the state variables with time" (Ižikevič, 2010, p.8). Therefore, although RNNs were only recently successful at learning complex tasks, RNNs as dynamic systems have received tremendous attention for a longer time (Guevara, Glass, Mackey, & Shrier, 1983; Sompolinsky et al., 1988; Wilson & Cowan, 1973). This line of research was critical in understanding how different variables impact the dynamics of RNNs and those findings are now crucial to understand how trained RNNs behave (Han, Shi, & Wang, 2004; Sussillo, 2014; Sussillo & Barak, 2013).

1.5 Neural dynamics of temporal processing

So far, we covered the behavioural and cognitive aspects of temporal processing. We also covered the different types of frameworks and

Model	Biological plausibility	Ability to learn temporal tasks	Millisecond timing	Second timing	Autonomous generation of activity	Continuous outputs
Delay-lines	High	Yes	Yes	No	No	No
Synfire chains	High	Yes	Yes	No	No	No
Pace-maker accumulators	Low	Yes	Yes	Yes	Yes	No
Beat frequency models	Medium	Yes	Yes	Yes	Yes	No
McCulloch-Pitts	Low	No	-	-	No	No
Perceptrons/ADALINE	Low	No	-	-	No	Yes
Multi-layer perceptrons	Low	No	-	-	No	Yes
Associative RNN	Low	No	-	-	No	Yes
RNN as dynamic systems	High	No	-	-	Yes	-
Long short-term memory	Low	Yes	No	No	No	Yes
Reservoir computing	High	Yes	Yes	Yes	Yes	Yes

Table 1: Temporal processing with the main models in computational neuroscience.

theories that can be used to model these tasks. In this section, we will cover the main findings in experimental neuroscience that might provide cues about how these functions are organized in the brain. As stated earlier, temporal processing includes a relatively broad range of tasks and inputs, and it isn't clear to what extent each of these heterogeneous processes relate to each other. It is therefore not surprising that we are still far from a general understanding of how the mechanisms of temporal processing are organized across the brain. In the next section, we will briefly cover the main brain regions that have been identified to play some role in temporal processing, and then we will discuss the underlying features of the neural activity of these regions.

1.5.1 *Localization of temporal processing*

Although still debated, most studies that investigated the localization of temporal processing in the brain seem to favour distributed systems, where multiple brain regions are either cooperatively or independently involved in some aspects of these functions (Ivry & Schlerf, 2008). We will cover the main brain structures that have been identified with lesion or neuroimaging studies. Although these methods

offer valuable knowledge about the localization of these functions, they offer little information on their neural mechanisms and computations. The greatest piece of evidence in favor of non-dedicated and distributed temporal processing systems are the findings of modality specific representations of time (Bueti, Bahrami, & Walsh, 2008; Johnston, Arnold, & Nishida, 2006; Karmarkar & Buonomano, 2007; Morrone, Ross, & Burr, 2005). Such findings suggest that each modality (visual, auditory, etc.) possess the circuitry required to represent and process temporal information. However, the temporal information contained in those peripheral circuits appears to be on the order of the millisecond timescale (Coull, Cheng, & Meck, 2011). This suggests that on the timescale of seconds, some sort of dedicated model of temporal processing could be involved (Buhusi & Meck, 2009; Ivry & Schlerf, 2008; Matell & Meck, 2000). There are two main brain regions involved as candidates for dedicated timing system, the cortico-striato-thalamic (CST) networks and the cerebellum (Ivry & Schlerf, 2008; Petter, Lusk, Hesslow, & Meck, 2016). We will first cover those two models of dedicated temporal processing, and we will then discuss the involvement of cortical networks in temporal processing. We will also cover the role of the monoaminergic systems, which provide support for both dedicated and intrinsic frameworks.

1.5.2 *The cerebellum*

The cerebellum was one of the first brain structures hypothesized to be involved in temporal processing. Ivry & Keele (1989) found that human subjects with lesions to the cerebellum had deficits in the production and perception of timing tasks. Neuroimaging studies also suggest a role for the cerebellum in temporal processing (Teki, Grube, Kumar, & Griffiths, 2011). It isn't clear what role the cerebellum plays

in temporal processing, as its primary role is often assumed to be related motor control (Doya, 2000). However, the cerebellum is suspected of being involved in interval timing from tens to hundreds of ms (Mauk & Buonomano, 2004). Experimental evidence suggests that the cerebellum might act as a general purpose interval timer, mostly for tasks involving delayed and trace conditioning (Kalmbach et al., 2010). This was first demonstrated in (Perrett, Ruiz, & Mauk, 1993), where rabbits were subjected to delayed eye-lid conditioning. After lesioning the cerebellum, the animals retained the learned response, but lost the ability to estimate the appropriate response timing. Some mechanisms behind this type of learning are fairly well understood (Ohyama, Nores, Murphy, & Mauk, 2003), and involve a feedforward architecture where the Purkinje cells of the cerebellum are integrating sensory inputs incoming from mossy fibres, and climbing fibres carry the error signal related to the estimated delay for a conditioned response (Medina & Mauk, 2000; Raymond & Medina, 2018). The temporal information is hypothesized to be generated by the granule cells, but the mechanisms that maintain this temporal information in memory after the presentation of the stimulus are still unknown (Medina & Mauk, 2000).

1.5.3 *Cortico-striato-thalamic networks*

The CST networks have received a great deal of attention regarding their implication in interval timing (Hinton & Meck, 2004; Meck & Malapani, 2004; Meck, Penney, & Pouthas, 2008; Paton & Buonomano, 2018; Rao, Mayer, & Harrington, 2001; Wittmann, Leland, & Paulus, 2007) and for shaping complex sequences of action (Jin & Costa, 2015). The basal ganglia is the main component of this temporal system. The striatum receives millions of projections from the cortical cells, and

is hypothesized to encode a starting time when a "go" cue is given (Buhusi & Meck, 2005) for a timing interval task. Such evidence comes mainly from fMRI studies that show that the striatum is activated during an interval categorization task (Rao et al., 2001). Although most studies on this circuit involve non-invasive methods such as EEG and fMRI, data collected with electrodes implants in vivo confirmed the involvement of these structures in temporal tasks. Mello, Soares, & Paton (2015) found that the striatum of rats performing interval timing tasks conveyed information relative to the time elapsed during the task and similar findings have been reported in the monkey's striatum (Chiba, Oshio, & Inase, 2015). Furthermore, direct manipulation of the striatal activity was found to alter duration judgments in rats (Gouvêa et al., 2015). There is also a lot of evidence in support of a coupling between the thalamus and the PFC during working memory tasks (Bolkan et al., 2017; Fuster & Alexander, 1973). Most studies on the role of the cortex in temporal processing tend to suggest distributed, modality specific mechanisms where motor (Kawai et al., 2015), auditory (Brosch, Selezneva, & Scheich, 2005) and visual cues (Chubykin, Roach, Bear, & Shuler, 2013; Shuler & Bear, 2006) produces prolonged neural responses that convey temporal information in their respective cortices. A recent study reported that in motor interval timing tasks, the motor cortex is necessary to learn the interval, but isn't involved once the interval has been learned (Kawai et al., 2015). This supports the CST view, where the cortices might play a role to train the basal ganglia, which acts as a population clock for temporal processing (Paton & Buonomano, 2018).

1.5.4 *Monoaminergic networks*

Dopaminergic (DA) neurons are an intrinsic part of the CST network, as the DA neurons from the substantia nigra compacta appear to be involved in the functions of this system. However, neurons from the ventral tegmental area (VTA) have also received tremendous attention due to their role in reward prediction error (RPE) (Schultz, 1998, 2016). RPE occurs when an unexpected reward is delivered to an animal, or when an unexpected cue signalling a future reward occurs. Given that there is usually a delay between a stimulus and the delivery of a reward, is it therefore not surprising that DA neurons would be involved in temporal processing. For instance, DA error signals increase as a logarithmic function of the duration of an interval between a stimulus and the delivery of reward (Fiorillo, Newsome, & Schultz, 2008), but overall decrease when a reward is associated with a longer delay (Kobayashi & Schultz, 2008). However, it isn't clear if the DA signal is directly involved in timing perception, as the manipulation of the VTA produces inconsistent results. For instance, up-regulating DA activity sometimes speed up the timing estimations (Buhusi & Meck, 2002; Maricq & Church, 1983), and sometimes does the opposite (Balci et al., 2010; Lake & Meck, 2013; Soares, Atallah, & Paton, 2016). Regardless of whether or not VTA neurons are directly performing the computations related to timekeeping, they seem to have an impact on temporal judgments (Soares et al., 2016). Other monoamines, such as norepinephrine (NE), acetylcholine (ACh) and serotonin (5-HT) have also been hypothesized to play a role in temporal processing, mostly in the context of eligibility traces (Huertas, Schwettmann, & Shouval, 2016). Eligibility traces are mechanisms of non-hebbian plasticity that have been proposed to reconcile the behavioural timescales of learn-

ing with the timescales of synaptic plasticity (Gerstner, Lehmann, Liakoni, Corneil, & Brea, 2018). Traditional hebbian mechanisms of long-term potentiation (LTP) and depression (LTD) operate on the scale of milliseconds (Bi & Poo, 1998; Markram, Lübke, Frotscher, & Sakmann, 1997), whereas the delay between a stimulus and a reward can be a few seconds long. Eligibility traces assume that synaptic activity encoding for the stimulus generates a short-lived physiological tag that can be converted in long-term plasticity upon receiving a delayed reward signal, often in the form of monoaminergic neurotransmitter release (Gerstner et al., 2018; Huertas et al., 2016), but not necessarily (see Bittner, Milstein, Grienberger, Romani, & Magee, 2017). One of the first experimental findings on eligibility traces was also a significant piece of evidence in favor of intrinsic distributed mechanisms of temporal processing (Shuler & Bear, 2006)). Shuler & Bear (2006) showed that cortical neurons in the V1 region would modulate their firing rate based on the expectation timing of the reward delivery. This finding suggested that the first stage of sensory cortices is somehow involved in temporal processing of sensory inputs. In a subsequent study, Chubykin et al. (2013) reported that such activity was necessary to learn (and potentially express) cue-reward intervals, and that this learning was ACh dependent. Additionally, they showed that delayed pulses of ACh was sufficient to extend the duration of the activity in V1, which offers strong evidence for the presence of eligibility traces in the cortex. Other studies have reported similar findings (Gerstner et al., 2018), including with other neurotransmitters and different brain regions such as NE and 5-HT in the prefrontal cortex (PFC) (He et al., 2015).

1.5.5 *Cortical and subcortical representations of time*

Different dynamics have been found to be underlying temporal processing in various brain networks. It was observed early on that some neurons, initially in the dorsolateral prefrontal cortex (DLPFC) (Fuster, 1973; Fuster & Alexander, 1971; Kubota & Niki, 1971) but also in other regions (S. Kastner, Pinsk, De Weerd, Desimone, & Ungerleider, 1999; Serences, Ester, Vogel, & Awh, 2009) showed persistent activity after the presentation of a stimulus. Synaptic reverberations (which falls under the RNN conceptualization of temporal processing) have been hypothesized to be the main mechanism behind to sustained activity of cortical circuits in the absence of external inputs (Wang, 2016). In this framework, recurrent excitatory loops within a network are necessary to self-sustain neural activity. One of the first direct pieces of evidence that supported this view came from Buonomano (2003) with recordings of cortical slices. This study showed *in vitro* that after a brief electrical stimulation, cortical slices would show self-sustained activity for up to a few hundreds of ms, and that this activity was driven by the networks recurrent connections. A subsequent study (Goel & Buonomano, 2016) found that this evoked neural activity could be trained in order to increase the length of its depolarization. Similar mechanisms as the monoaminergic formation of eligibility traces (Chubykin et al., 2013) might be involved in this process. In summary, there are well established neural features of temporal processing which shows sustained activity across many brain regions. In the next section we will cover the main interpretations of such activity and what kind of information it is hypothesized to encode.

1.5.6 *Ramping and bistability*

One way to explain the sustained activity in brain circuits is with the attractor networks described previously (Amit & Brunel, 1997). These models stipulate that if the recurrent excitatory connections within a population of neurons are strong enough, they can switch to up-states, thus showing bistable modes with down states. This model offers a compelling explanation of how items can be stored in working memory, where a sub-population of the network encoding for an item gets activated when the item is memorized. This activity allows the item to be recalled after a delay (Wang, 2016). Such dynamics are commonly referred to as "bump attractors" (Wimmer, Nykamp, Constantinidis, & Compte, 2014). These self-sustained attractors can explain how circuits of neurons can maintain information for which they are tuned to in memory without the presence of an external input. There is an overwhelming amount of empirical evidence supporting this model in primates with visual memory tasks (Constantinidis, Franowicz, & Goldman-Rakic, 2001; Funahashi, Bruce, & Goldman-Rakic, 1989), in rodents for head direction cells (Taube & Muller, 1998; Zugaro, Arleo, Berthoz, & Wiener, 2003) and in drosophila for head direction cells (Kim, Rouault, Druckmann, & Jayaraman, 2017). This persistent firing of cortical circuits in the absence of external stimulation can be highly time-varying. This feature of neural dynamics is often ignored in the literature (Brody, Romo, & Kepecs, 2003), although it was already noticed in the first neurophysiological studies on working memory (Brody, Hernández, Zainos, & Romo, 2003; Fuster & Alexander, 1973; Kubota & Niki, 1971). Among those features of neural activity are ramping dynamics, where a neuronal population linearly or non-linearly increases its firing rate in

time. Many models of interval timing suggest that the slope of linear ramping activity is modulated by the length of a timing interval (Janssen & Shadlen, 2005; Leon & Shadlen, 2003), suggesting that this accumulation of activity encodes elapsed time. There are many additional roles that have been proposed for the linear and non-linear dynamics of bistable cortical circuits, where this activity would not explicitly encode temporal information. For instance, such dynamics have been linked to accumulator models for decision-making (Berberian, MacPherson, Giraud, Richardson, & Thivierge, 2017; Roitman & Shadlen, 2002), where the activity of some neurons can both linearly and non-linearly evolve to represent evidence in favor of a decision. Such activity might also simply be a by-product of other inputs. In the PFC, task related up-states co-exist with decision-making, motor planning, reward coding and anticipation, which could all alter networks which could otherwise evolve toward a steady-state (Brody, Romo, & Kepecs, 2003). Although very useful for discrete tasks such as categorization and recognition, it isn't straightforward how the dynamics of bistable networks could generalize to continuous tasks. That is because even if dynamic activity has been associated with bump attractors, they are more readily applicable to produce discrete responses, and not the precise patterns of neural activity required for continuous tasks (such as motor control, speech production or interval timing).

1.5.7 *Dynamical attractors*

As discussed previously, SDC requires dynamic attractors where neural trajectories evolve in time, as opposed to other types of networks which requires the activity to settle to a given steady-state. Therefore, networks of neurons have to exhibit transient dynamics related to the task that they execute. Such dynamics have to be structured and re-

peatable across trials. Experimental evidence for this type of activity was found in the zebrafish (Friedrich & Laurent, 2001) and the locust (Laurent & Davidowitz, 1994) in olfactory detection experiments. These experiments showed that presenting odours to these organisms produces trajectories of neural activity, and each trajectory encodes a different stimulus. This idea of discriminating different stimuli by separating their neural activity over time was initially shown to be computationally efficient by Buonomano & Merzenich (1995). Early experiments attempting to detect neural activity to support this form of computations were limited by the number of neurons that could be simultaneously record from. Despite this limitation, Seidemann, Meilijson, Abeles, Bergman, & Vaadia (1996) were able to record a handful of isolated neurons recorded from the frontal cortex of monkeys performing a delayed localization task. They were able to identify transitions between multiple states in the population activity during the delay across trials. Those delays were consistent with the monkey's response, and that suggests that these fluctuations were conveying task-related information during the delay. With advances in multiple neurons recording techniques, a deluge of evidence supporting repeatable spatio-temporal patterns of activity in the cortex began in the mid 2000s. Baeg et al. (2003) found that ensembles of neurons in the PFC were sequentially activated in working memory tasks. Similar findings were made in the PFC (Bakhurin et al., 2017; Bolkan et al., 2017; Fujisawa, Amarasingham, Harrison, & Buzsáki, 2008), the hippocampus and enthorinal cortex (Heys & Dombeck, 2018; Pastalkova, Itskov, Amarasingham, & Buzsáki, 2008), the parietal cortex (Harvey, Coen, & Tank, 2012; Runyan, Piasini, Panzeri, & Harvey, 2017), the motor cortex (Peters, Chen, & Komiyama, 2014), the striatum (Bakhurin et al., 2017; Mello et al., 2015; Sheng, Lu, Shen, & Poo, 2019) and HVC

neurons of the songbird (Katlowitz, Picardo, & Long, 2018). It is now well established that cortical and sub-cortical populations generate task-related patterns of activity, and that this activity can be decoded by experimenters to predict the actions of the animal (Bakhurin et al., 2017; Seidemann et al., 1996). Despite the abundance of experimental reports on sustained task-related activity, we know very little about the mechanisms at its origin.

1.6 Outline of this work

We've covered the main aspects of temporal processing in the brain, as well as the main competing theories attempting to explain the behaviourally and neurologically data related to this function. Reservoir computing is one of the most recent theories of how RNNs can be trained to produce complex temporal tasks. However, there are three main limitations to their current quality as a model of brain networks. Chaos in RNN has been extensively studied (Brunel, 2000; Lajoie, Lin, & Shea-Brown, 2013; London, Roth, Beeren, Häusser, & Latham, 2010; Sompolinsky et al., 1988; C. van Vreeswijk & Sompolinsky, 1996; C. v. van Vreeswijk & Sompolinsky, 1998). Chaos is defined as sensitivity to small differences in initial conditions, which in the perspective of dynamic systems refers to how infinitesimal perturbations to the state of the network are impacting its activity. However, we know little about the impact of infinitesimal perturbations on the laws governing those states and their impact on the network dynamics. An important component of those laws in RNNs are the connections between neurons. One would expect that rewiring or removing a small number of synaptic connections, such as rewiring 1% of the synapses, should not have a catastrophic impact on the network dy-

namics and learning (Brody, Romo, & Kepecs, 2003). In the second chapter, we show that RC using chaotic networks that are tolerant to perturbations to their state (Laje & Buonomano, 2013; Sussillo & Abbott, 2009) are extremely sensitive to structural perturbations, and that removing a single neuron out of thousands is enough to completely disrupt a learned task. We explore how different types of connectivity impact the robustness to such perturbations, and we introduce a fundamental modification to the architecture of the model that dramatically increases its resilience to structural disruptions. We do so by removing the feedback pathway from the readout to the reservoir, and using oscillatory inputs to tame the chaotic activity of the network. In the third chapter, we perform a more thorough investigation of the oscillation driven model introduced in the second chapter. We used more realistic and biologically relevant reservoir of spiking units with conductance based synapses. We show that oscillatory inputs provide a powerful mechanism to autonomously generate aperiodic neural activity so the network can learn any arbitrary time-varying output function. We show that the oscillations can be endogenously generated by spiking networks with parameters similar to the reservoir. This model is able to learn to perform motor control tasks such as drawing shapes and is able to generate speech with different rescaling factors. Finally, we show that this network is able to bind sequences of discrete events together, which provides a theoretical account of theta sequences recorded in the hippocampus during spatial navigation.

2 CHAPTER 2: DRIVING RESERVOIR MODELS WITH OSCILLATIONS: A SOLUTION TO THE EXTREME STRUCTURAL SENSITIVITY OF CHAOTIC NETWORKS.

A large body of experimental and theoretical work on neural coding suggests that the information stored in brain circuits is represented by time-varying patterns of neural activity. Reservoir computing, where the activity of a recurrently connected pool of neurons is read by one or more units that provide an output response, successfully exploits this type of neural activity. However, the question of system robustness to small structural perturbations, such as failing neurons and synapses, has been largely overlooked. This is in contrast to well-studied dynamic perturbations that lead to divergent network activity in the presence of chaos, as is the case for many reservoir networks. Here, we distinguish between two types of structural network perturbations, namely "local" (e.g. individual synaptic or neuronal death) and "global" (e.g. network-wide fluctuations). Surprisingly, we show that while global perturbations have a limited impact on the ability of reservoir models to perform various tasks, local perturbations can produce drastic effects. To address this limitation, we introduce a new architecture where the reservoir is driven by a layer of oscillators that generate stable and repeatable trajectories. This model outperforms previous implementations while being

resistant to relatively large local and global perturbations. This finding has implications for the design of reservoir models that capture the capacity of brain circuits to perform cognitively and behaviourally relevant tasks while remaining robust to various forms of perturbations. Further, our work proposes a novel role for neuronal oscillations found in cortical circuits, where they may serve as a collection of inputs from which a network can robustly generate complex dynamics and implement rich computations.

2.1 Introduction

A fundamental building block of cognition and behaviour is the ability of brain circuits to maintain information about objects beyond their time of appearance in the environment. Experimental evidence suggests that sustained patterns of neural activity may constitute a simple correlate of this ability. Indeed, patterns of sustained activity are well-known to guide behaviour in tasks including motor control (Churchland & Shenoy, 2007; De Zeeuw et al., 2011), interval timing (Goel & Buonomano, 2014) and delayed discrimination (Mazurek, Roitman, Ditterich, & Shadlen, 2003; Romo, Brody, Hernandez, & Lemus, 1999). Going further, sustained activity encompasses a broad range of neural phenomena and is not limited to cases where neurons maintain a fixed firing rate over time. In fact, patterns of sustained activity may take the form of precise fluctuations in firing rate in time (Buonomano & Maass, 2009) or the timing of spike patterns (Mainen & Sejnowski, 1995; Tiesinga et al., 2008). However, it is unclear how neural circuits generate reliable patterns of neural activity given that the spike train produced by a neuron is highly dependent on its pre-stimulus history

(Arieli, Sterkin, Grinvald, & Aertsen, 1996; Banerjee, Seriès, & Pouget, 2007) and that the addition of a single spike in a train of action potentials can cause significant post synaptic variations (London et al., 2010) (c.f. Lajoie et al., 2013). One framework that has been successful at capturing patterns of temporal activity is reservoir computing (Jaeger, 2002; Maass et al., 2002), and has been proposed as a plausible theory of high-level cortical information processing. The core principle of reservoir computing consists in mapping the states of neurons through time (termed neural trajectory) in a recurrent circuit to a specific output function by adjusting the connection weights of one or many downstream units (termed readout units). A body of work suggests that reservoir computing can account for neural dynamics in different regions of the brain (Barak et al., 2013; Toledo-Suárez, Duarte, & Morrison, 2014; Yamazaki & Tanaka, 2007). Furthermore, there is increasing evidence suggesting that principles of reservoir computing could explain cognitive and behavioural processes (Barak et al., 2013; Bernacchia, Seo, Lee, & Wang, 2011; Enel, 2014; Joshi & Maass, 2005; Laje & Buonomano, 2013; Mante, Sussillo, Shenoy, & Newsome, 2013; Sussillo & Abbott, 2009; Sussillo et al., 2015) One interesting characteristic of some reservoir models is their ability to generate activity as autonomous systems (that is, systems whose ongoing dynamics are not driven by fluctuating inputs, c.f. Lajoie et al., 2013; Lajoie, Thivierge, & Shea-Brown, 2014). This is achieved by adjusting the parameters of the reservoir to place it in a chaotic state where it generates rich spontaneous activity (see e.g. Sompolinsky et al., 1988). This regime has been proposed as a possible mechanism underlying sustained irregular activity observed in cortical areas (C. van Vreeswijk & Sompolinsky, 1996). Such activity has been harnessed in some models to create stable and repeatable trajectories in the absence of ongoing external

stimulation (Hoerzer, Legenstein, & Maass, 2014; Laje & Buonomano, 2013; Sussillo & Abbott, 2009; Sussillo et al., 2015). In these models, chaos is suppressed either through strong feedback from the readout units in the reservoir or by stabilizing some neural trajectories by adjusting connection weights of the reservoir. These models are robust to the addition of noise (Laje & Buonomano, 2013) as well as small alterations in network connectivity (Sussillo & Barak, 2013; Sussillo et al., 2015). Therefore, these reservoirs models are attractive candidates to explain both the origin of stable patterns of neural activity and how such activity can be exploited by downstream neurons in a meaningful way. However, despite the large body of work lying parallels between neural networks in the brain and reservoir computing systems, an important biological constraint has been overlooked: the resistance to small structural perturbations such as failures in synaptic transmission or neuronal death. Such perturbations frequently occur in healthy circuits in the brain without leading to drastic functional changes (Fu, Yu, Paxinos, Watson, & Rusznák, 2015; Pakkenberg et al., 2003; Westlye et al., 2009). This is what we investigate in this work. We begin by testing the resistance of different reservoir models (Laje & Buonomano, 2013; Sussillo & Abbott, 2009) to post-training perturbations, while drawing a distinction between structural (parameters alteration, such as neuronal and synaptic loss) and state-space (alteration of the firing rate of the neurons) perturbations. Further, we examine the impact of both local and global forms of structural perturbations. On the one hand, global perturbations are noisy fluctuations in the activity of multiple neurons, or random variations of synaptic strengths across the network that are analogous to the changes that occur under synaptic plasticity (Froemke & Schreiner, 2015). On the other hand, local perturbations consist of failure of individual

synapses or neurons. Here, we show that local and global perturbations of similar magnitude have a markedly different impact on the dynamic of recurrent neural circuits. Going further, some perturbations are more devastating than previously reported (Laje & Buonomano, 2013; Sussillo & Barak, 2013; Sussillo et al., 2015). In fact, we show that minimal perturbations can completely impair the capacity of the network to execute some previously learned tasks. Several solutions may address the issue of instability to structural perturbations in chaotic networks. For instance, it is possible to suppress chaos in spontaneously active networks of recurrent firing rate units by the injection of white noise (Molgedey et al., 1992) or time-varying input signals (Bertschinger & Natschläger, 2004; Rajan, Abbott, & Sompolinsky, 2010). However, these solutions assume that such input is available alongside the network during the execution of the task. As an alternative, we examined a solution where a layer of input units to the network behave as oscillators. We show that this model can generate robust trajectories and exhibits resilience to perturbations. This new framework is simpler, more efficient, and less training intensive than previous models.

2.2 Methods

2.2.1 *Architectures*

In this work, we employ networks of firing rate units connected according to five possible architectures. Architectures A,B and C have been proposed in (Sussillo & Abbott, 2009) and we refer to their implementation as standard FORCE; architecture D is proposed in (Laje & Buonomano, 2013) and we refer to its implementation as innate

training. We introduce architecture E in this work and refer to its implementation as the driven model. The various architectures (A to D, Fig. 1; E, Fig. 10(a)) can be summarized as follows:

(A) Only the synapses from the reservoir to the readout unit are modified during training and there is direct feedback from the readout unit to the reservoir.

(B) The feedback is provided by a control network instead of the readout unit. The synapses from the reservoir to the readout unit and to the control network are modified during training.

(C) No external feedback is provided to the reservoir. Rather, the reservoir is its only source of feedback and is modified during the training phase. This is achieved by applying the same learning rule to the readout unit and to the neurons in the reservoir.

(D) A subset of the neurons in the reservoir is adjusted to reinforce an innate trajectory of the network. This trajectory is triggered by kicking the network in a delimited state-space with an external input. Once this process is complete, the synapses of the readout unit are modified.

(E) A layer of oscillators controls the trajectory of the reservoir based on their relative phase. Only the synapses from the reservoir to the readout unit are modified during training.

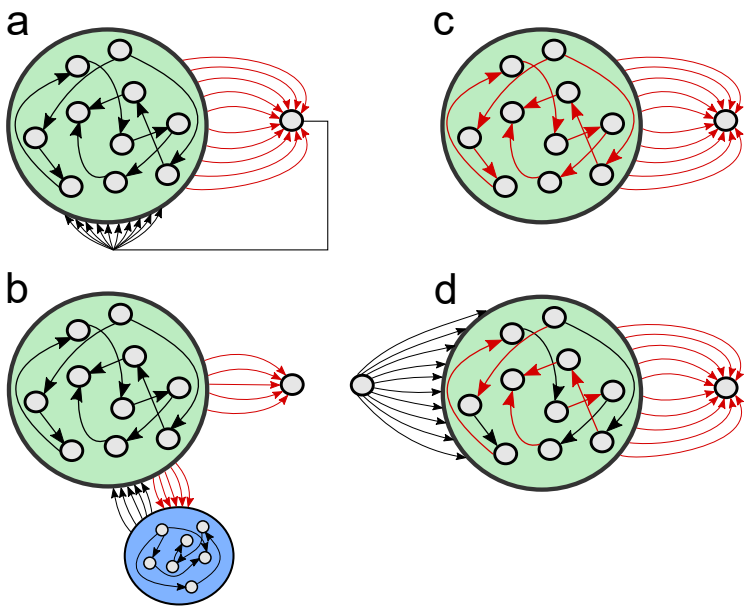

Figure 1: Schematic representation of network architectures. Connections in black are randomly initialized and left untouched during training whereas connections in red are modified according to the training rule. **a** No external inputs are used to control the activity of the reservoir (green). Instead, the network relies on strong feedback from the readout unit to the reservoir to control the chaotic activity. **b** This architecture uses a control network (blue) instead of the readout unit to provide feedback to the reservoir. **c** Learning occurs in the reservoir as well as on the connections of the readout unit. The reservoir is its only source of feedback. **d** The reservoir displays chaotic spontaneous activity in the absence of input. After a brief pulse is injected in the network, the reservoir follows a trajectory that has been previously stabilized by training a subset of neurons.

2.2.2 *Neural activity and parameters*

We kept the same parameters used in most simulations of the original articles describing FORCE and innate training. Both models are described by the following equations:

$$\tau \frac{dx_i}{dt} = -x_i + \sum_{j=1}^{N_R} W_{ij}^R r_j + \sum_{\mu=1}^{N_I} W_{i\mu}^{RI} I_\mu, \tag{1}$$

$$z = \sum_{j=1}^{N} W_{ij}^{zR} r_j, \tag{2}$$

where $r_i = \tanh(x_i)$ (i.e., hyperbolic tan function) is the firing rate of the recurrent units $x_i (i = 1, ..., N_R)$, $I_\mu (\mu = 1, ..., N_I)$ represents the input units, z represents the readout unit and the time constant $\tau = 10ms$. The reservoirs' weight matrices W^R are of size $N_R \times N_R$ where the non-zero values were drawn from a Gaussian distribution with a mean of zero and standard deviation equal to $g_R/\sqrt{p_R N_R}$, where g_R is the gain of the synaptic weights and p_R is the proportion of non-zero values in the matrices. Weight matrices connecting the input unit to the reservoir (W^{RI}) and connecting the reservoir to the readout units (W^{zR}) were initialized differently depending on the architecture. *Standard FORCE.* The parameters used with architecture A, B and C can be found in Table 2. Readout weights W^{zR} were initially set to zero and when feedback from the readout unit z to the reservoir was used, the values of W^{Rz} were drawn from a uniform distribution with values between -1 and 1. No external inputs were used for our simulations so no weights W^{RI} were required. Architectures A and B require the addition of some components to equation 1:

$$+ W_i^{R_z} z, \tag{3}$$

$$+ \sum_{a}^{N_F} W_{ia}^{RF} s_a, \tag{4}$$

$$\tau \frac{dy_a}{dt} = -y_a + \sum_{b=1}^{N_F} W_{ab}^{F} s_b + \sum_{i=1}^{N_G} W_{ai}^{FR} r_i, \tag{5}$$

The activity of architecture C is described with equation 1. The addition of equation 3 to equation 1 results in architecture A. The addition of equation 4 to equation 1 results in architecture B. Equation 5 provides the dynamics of the control network where $s_a = \tanh(y_a)$ is equal to the firing rate of $y_a (a = 1, ..., N_F)$. The control network's weight matrix W^F is of size $N_F \times N_F$ where the non-zero values were drawn from a Gaussian distribution with a mean of zero and standard deviation equal to $g_F/\sqrt{p_F N_F}$, where g_F is the gain of the synaptic weights and p_F is the proportion of non-zero values in the matrix. Connections from the reservoir to the control network W^{FR} were initially set to zero and the non-zero values of W^{RF} were drawn from a uniform distribution with values between -1 and 1. *Innate training.* This architecture is shown on Fig. 1(d) and is described by equations 1 and 2. The default parameters used in architecture D can be found in Table 1. The input weights W^{RI} were drawn from a Gaussian distribution of mean zero and unit standard deviation and the values of W^{zR} were drawn from a Gaussian distribution with mean of zero and standard deviation of $1/\sqrt{N}$. The value of the active input unit for a given trial was held at zero except at the time of stimulation where it had an amplitude of 5 (arbitrary units) for 50 ms. *Driven model.* The architecture E is shown in Fig. 10(a) and the parameters of this model are the same used for innate training (architecture D), except for a different input pattern and the absence of training of the reservoir's connections. The input layer is made of distinct oscillating units of different frequencies and a constant amplitude. The initial phase of the oscil-

Table 2: Parameters used for the different architectures.

	Architectures			
	A	B	C	D
Reservoir size (N_R)	1000	2000	1000	1000
Number of input units (N_I)	0	0	0	1
Learning rate (α)	1	1	1	1
p_R	0.1	0.1 ($P_F = 1$)	1	0.1
p_{zR}	1	0.5 ($P_{FR} = 0.5$)	1	1
p_{Rz}	1	0 ($P_{RF} = 0.5$)	0	0
g_R	1.5	1.5 ($g_F = 1.2$)	1.5	1.5

lators was chosen randomly for all simulations, and unless specified, their frequencies were randomly selected from a uniform distribution with a minimum of 1 Hz and a maximum of 5 Hz. Non-zero values of the input weights were drawn from a Gaussian distribution with a mean of zero and standard deviation equal to $g_{RI}/(N_I p_{RI})$, where g_{RI} is the gain of the input weights, N_I is the number of input units (oscillators) and p_{IR} is the proportion of non-zero values in the connection matrix. The default values of g_{RI} and p_{RI} are of 1.5 and 0.5, respectively.

2.2.3 *Training tasks*

Two different tasks were employed. In a first task, a signal composed of four sinusoidal waves was used as the target output. This signal was made of 12 cycles generated with sinusoids of same period but different amplitudes (Fig. 2(a)). We trained architectures A-C on this task, and assessed the performance of networks using the mean absolute error (MAE) between the output and the target values. In a second task, termed the timing task, the readout target was determined by $f(t)$, set to a constant value of 0.2 until the start of the

Gaussian curve with a peak centered at time t_{delay} of 1 second. The target trajectory was collected from the activity of the network from the end of the input pulse to the end of the Gaussian curve of the target function (1,150 ms after the end of the input pulse). Three different methods were used to assess the performance of the networks. The mean squared error (MSE) was used to compare the output and the target of the readout unit. We also computed the interval timing estimated by the network based on the time where it crossed a threshold and compared it to the actual peak of the target to obtain a timing lag. Finally, we computed the Pearson correlation coefficient between the target and the output. Even though the original model tolerates a rather large amount of external noise, we kept noise off across all of our simulations to isolate the effect of the perturbations under study. All error bars reported in the results below represent the standard error of the mean.

2.2.4 *Learning algorithms*

We used a powerful algorithm termed first-order reduced and controlled error (FORCE) learning (Sussillo & Abbott, 2009) that can train networks to reproduce periodic patterns (e.g., a complex sinusoid) or chaotic signals for limited time intervals (e.g., a Lorenz attractor). This algorithm is based on the recursive least squares method (Haykin, 2002), and was used to train all architectures, but the subset of synapses trained as well as the target function differed between architectures. All architectures had the following equation to adjust the readout weights:

$$W^{zR}(t) = W^{zR}(t - \Delta t) - e(t)P(t)r(t), \tag{6}$$

$$e(t) = W^{zR^T}(t)r(t) - f(t). \tag{7}$$

Where the error $e(t)$ was determined by the difference between the value of the readout unit obtained with the multiplication of the reservoir's activity with the weights W^{zR}, and the target function's value f at time t. Each weight update was separated by a time interval Δt of 2 ms for all simulations. P is a running estimate of the inverse of the correlation matrix of the network rates r, modified according to equation 8 and initialized with equation 9.

$$P(t) = P(t-\Delta t) - \frac{P(t-\Delta t)r(t)r^T(t)P(t-\Delta t)}{1 + r^T(t)P(t-\Delta t)r(t)}, \tag{8}$$

$$P(0) = \frac{I}{\alpha}. \tag{9}$$

where I is the identity matrix and α is a learning rate constant. For architecture C, where the reservoir is subject to training, the whole connectivity matrix of the reservoir is modified with equation 6. For innate training, the same version of FORCE learning described above was employed to train the reservoir as well as the readout unit. However, innate training requires an independent estimate of the inverse correlation matrix of the presynaptic inputs P for each plastic neuron in the reservoir:

$$W_i^R(t) = W_i^R(t-\Delta t) - e_i(t) \sum_{k \in B(i)} P_{jk}^i(t) r_k(t), \tag{10}$$

$$e_i = r_i(t) - R_i(t), \tag{11}$$

$$P_{jk}^i(t) = P_{jk}^i(t-\Delta t) - \frac{\sum_{m \in B(i)} \sum_{n \in B(i)} p_{jm}^i(t-\Delta t) r_m(t) r_n(t) P_{nk}^i(t-\Delta t)}{1 + \sum_{m \in B(i)} \sum_{n \in B(i)} r_m(t) P_{mn}^i(t-\Delta t) r_n(t)}. \tag{12}$$

where $B_i(i = 1, ..., N_{Plas})$ is the subset of presynaptic neurons in the reservoir (60% of the reservoir) and equation 11 represents their error on unit i. The term e_i is the error based on the discrepancy between $r_i(t)$ (activity of neuron i) and $R_i(t)$, the target activity based on the innate trajectory. Equation 12 is an altered version of equation 8 where each plastic unit (only plastic units have their synapses adjusted) has an independent matrix P, which are identity matrices initialized according to equation 9 (with a size equal to the number of presynaptic units k). The innate training of the recurrent weights was applied for 20 iterations of the reservoir trajectory, then the weights of the readout unit were trained for 10 iterations of the target output. The training started at the end of the input pulse until the end of the Gaussian peak of the target. As with FORCE learning, each weight update was separated by a time interval Δt of 2 ms for all simulations.

2.2.5 *Reservoir dynamics*

The reservoir dynamics were quantified across three different metrics that can be used as indicators of their quality for learning. The *largest lyapunov exponent* (LLE) is an indicator of chaotic activity, where values larger than 1 indicates that infinitesimal perturbations will lead to exponential differences in trajectories. The *network saturation* can be detrimental to learning tasks because the network's unit tend to spend more time on both their minimum and maximum values. Finally, the *network correlation* is an indicator of how much the activity of each cell in the reservoir is correlated with the other cells, which leads to redundancy and impoverish the overall network dynamics.

Largest Lyapunov Exponent

The LLE was computed numerically, following the methods of (Laje & Buonomano, 2013). However, we didn't perform a manual selection for the intervals where the log(distance) versus time plot has a linear slope, because of the high number of tests we had to perform. For every condition we used a predetermined fixed interval (from 100 to 400 ms).

Network saturation

As the overall activity of the network increases, units tend to saturate due to the sigmoid function. This is undesirable because some values of the neuron's activity tend to be overrepresented (mostly at -1 and 1). We developed a measure based on the entropy of individual neurons of the network normalized by the maximum possible entropy value. We discretized the range of possible values (-1 to 1) in K bins, and for each neuron, and we plotted the amount of time the neuron spent in each bin during a simulation of length S. We then applied the entropy measure H on each neuron n_i (i = 1, ..., N), defined by:

$$H_n = -\sum_{j=1}^{K} p_j log p_j, \tag{13}$$

where K is the number of bins and is set to 20, and p_j is the probability that the neuron's value is in the range of bin j during the simulation. Then, H is normalized with the maximal entropy, which is the entropy of a discrete uniform distribution [-1, 1], n = K, as follow:

$$H_{norm} = \frac{H_{max} - H_n}{H_{max}}, \tag{14}$$

where $H_{max} = \log(n)$. Then, the network saturation is obtained by averaging H_{norm} across all neurons,

$$H_{network} = \frac{\sum_{i=1}^{N} H_{norm_i}}{N}. \tag{15}$$

Network correlation

The Pearson correlation was applied to a sample of pairs of neurons in the network. Ten percent of all pairs were taken. For each pair, the Pearson correlation was applied on the activity between both neuron's activity during a simulation of length S. Then, the correlation between all the selected pairs was computed and then averaged to obtain the network correlation.

2.3 Results

We begin by showing the impact of the different types of perturbation on the performance of two training schemes described in Methods (standard FORCE and innate training). First, we show that local perturbations have a marked impact on the trained models. Then, we introduce an alternative model that can overcome the issue of robustness while increasing the performance of the network on a task of interval timing.

2.3.1 *Structural perturbations in standard FORCE*

As a starting point, we examined how small structural alterations can impact the behaviour of a network with standard FORCE learning (Sussillo & Abbott, 2009). The first type of perturbation that we investigated is the clamping of individual neurons. This was achieved by setting the activation as well as all afferent and efferent connections

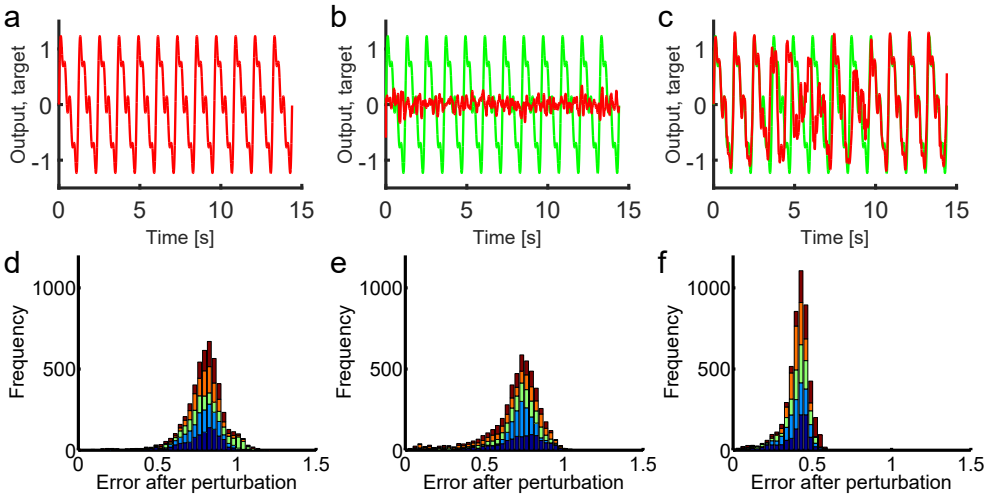

Figure 2: Activity of intact and damaged networks. Top: the output signal of a network is in red and the target sinusoid is in green. **a** The output of a network once the training phase has ended is almost perfectly overlapping with the target. **b,c** represent the outputs of selected networks with N-1 neurons of architectures A (MAE: 0.62 and 0.40). **d,e,f** show the distribution of MAE obtained from 5,000 trials with architecture A, B and C, respectively. The average MAE are of 0.79, 0.69 and 0.41, respectively. When normalized with the amplitude of the output function, the averages are of 0.61, 0.53 and 0.56. Each color represents the distribution of a different network, from which each neuron of the reservoir was clamped in turn. Each network has an initial MAE below 0.01.

of a given subset of neurons to zero. We began by training a network with each architecture (Fig. 1(a-c)) to replicate a jagged sinusoidal signal. We used the MAE between the output and the target in order to evaluate training performance. Once the network was trained (a fixed number of cycles of the sinusoidal pattern, here 12 cycles), all architectures attained a similar level of error on a distinct testing phase of the same length (Fig. 2(a)) but without further modifications (for 100 trials, architecture A: average MAE of 0.06, SEM of 0.01, architecture B: average MAE of 0.06, SEM of 0.01 and architecture C: average MAE of 0.05, SEM of 0.01). However, only networks with a MAE below 0.01 (arbitrary threshold to ensure only the best models are included for subsequent perturbation analysis) were selected for further analysis. In Fig. 2(b), we clamped the activity of one random neuron from the reservoir (out of 1,000 neurons) to zero from architecture A, and left the readout neuron untouched. We then allowed the network to try and produce the jagged sinusoidal signal without further training. On most trials, this small perturbation produced catastrophic disruptions in network activity. In some rare cases, the networks were able to retain some of their performance (Fig. 2(c)), but were still unstable. All three architectures were highly vulnerable to the clamping of a single, randomly selected neuron (out of a total of 1,000, representing 0.1% of all neurons in the reservoir). To further show this effect, we calculated the average MAE on multiple trials for each architecture (Fig. 2(d-f)). For each of the three architectures tested (A,B and C), five networks were generated (represented by different colors) from which the activity of a different neuron was clamped to 0 on each trial, for a total of 1,000 trials per network (one for each neuron). The average MAE over the 5,000 trials with architectures A, B and C were of 0.79, 0.69 and 0.41, respectively. The amplitude of the target function of architecture

C was almost halved compared to the two others to stay consistent with (Sussillo & Abbott, 2009) for fair comparison. That explains the shift of the distribution of Fig. 2(f) toward lower values. Because the MAE is dependent on the amplitude of the target function (greater amplitude leads to higher values of MAE), we normalized the average error based on this value (i.e., average MAE divided by the amplitude of the target). This yielded the following normalized average MAE: 0.61, 0.53 and 0.56 for architectures A-C, respectively. After normalization, the performance of architecture C was not significantly different from architecture A and B. In summary, all architectures used with the standard FORCE learning procedure were highly vulnerable to the clamping of a single neuron. Next, we examined the sensitivity of networks with innate training. Even though this model is also trained with the FORCE algorithm, it relies on different principles to generate stable reservoir trajectories.

2.3.2 *Structural perturbations in innate training*

We tested innate training models on a timing task where the network was taught to remain quiescent and to peak after an interval of one second (Fig. 3(a-d)). Once the reservoir's trajectory was trained for 20 iterations and the readout unit was trained for 10 iterations, the network's output had an almost perfect fit with the target function (Fig. 3(a)). We then clamped different subsets of neurons within the reservoir and computed the MSE between the output and the target of the readout unit. The trained networks lost their ability to output a stable value between the input pulse and the target peak after a single neuron was clamped in the reservoir (Fig. 3(b)). Fig. 3(a-d) shows the output of a trained network on five different trials (red curves) where a different subset of neurons (Fig. 3(b-d), number of neurons of 1, 2

and 5, respectively) has been clamped. Nevertheless, some networks could produce a peak in their output after the appropriate interval despite the perturbation. We then calculated the interval timing based on the time when the output crossed a threshold. In order to avoid penalizing a shift in the amplitude of the output value after the perturbation, we computed an independent threshold for every perturbation for each network. To find the most appropriate thresholds, we computed the average delay for every trial of a condition with a threshold that was increased from 0 to 1 (maximal value of the target function) with increments of 1e-03. The threshold associated with the best performance was selected. We also computed the success rate, where trials were considered successful when they fell within a 40 ms window centered on the target's peak. The performance degraded quickly as more neurons were clamped in the reservoir (Fig. 3(e,f)). The error values started to plateau when a little more than 1% of the neurons were clamped. The MSE between the output and the target increased monotonically as further neurons were eliminated (Fig. 3(f)). In addition to clamping entire subsets of neurons, we verified the impact of setting the weights of individual synapses to zero in the network (Fig. 4(a-c)). On the timing task, damaged networks were near the maximal error value when 1% of the synapses were clamped (Fig. 4(a)). The performance curve of networks with perturbed reservoirs with 0% to 1% of synapses set to zero was similar to the performance of networks from which 0% to 100% of the neurons connecting the reservoir to the readout unit were set to zero (Fig. 4(d-f)). This shows that in terms of proportion, the readout synapses are much less important for the performance of the network than the reservoir's synapses, as long as the trajectory in the reservoir remains intact. We also tested the impact of Gaussian structural disruptions distributed across all

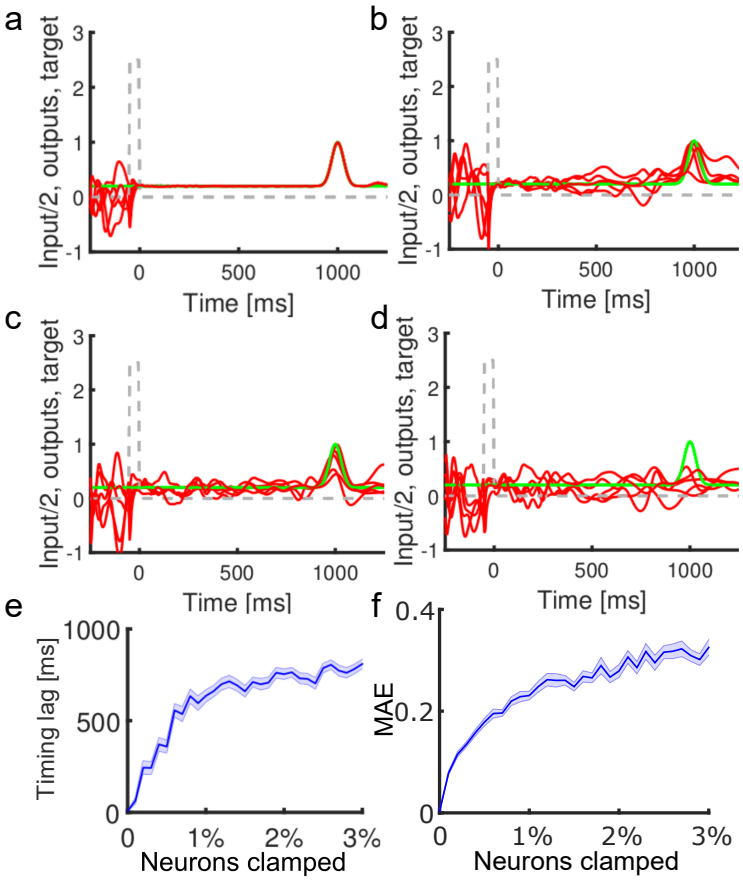

Figure 3: Performance of innate training on the timing task after perturbations. Each reservoir has a size of 1,000 neurons. An external input (grey dashed line) act as a reset on the network's state, which triggers the trajectory previously stabilized in the reservoir. The network is trained to peak 1,000 ms after the end of the input signal. **a,b,c,d** show the outputs of networks with N, N-1, N-2 and N-5 neurons in the reservoir, respectively. **e** Lag between the time the output crossed the threshold and the peak of the target. **f** MAE between the output and the target signal. Shaded areas represent the standard error of the mean.

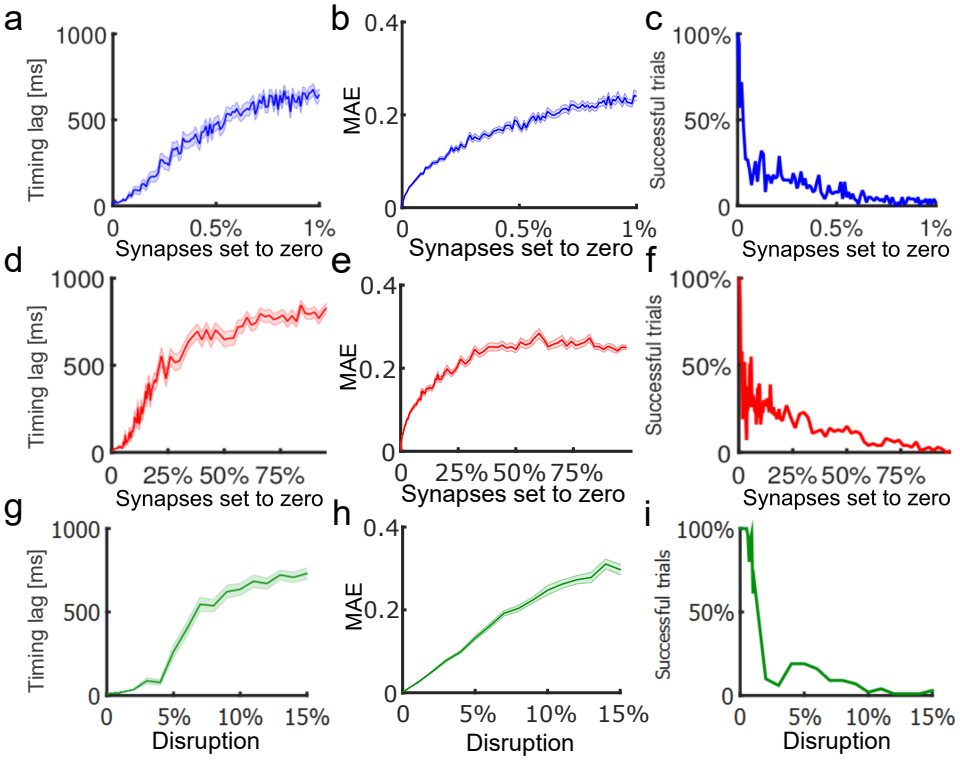

Figure 4: Performance on the timing task after different types of perturbations. Impact of clamping individual synapses of the reservoir and of the readout unit, as well as a Gaussian perturbation on all synapses in the reservoir. **a,b,c**: disabling synapses from the reservoir. **d,e,f**: disabling synapses that connect the reservoir to the readout. **h,i,j**: disruption of synaptic weights. The leftmost column of the figure shows the lag between the estimated timing of the output and the target. The second column shows the MAE between the output and the target. The rightmost column shows the proportion of trials for which the timing estimate fell within a 40 ms window centered on the target's peak. Shaded areas represent the standard error of the mean.

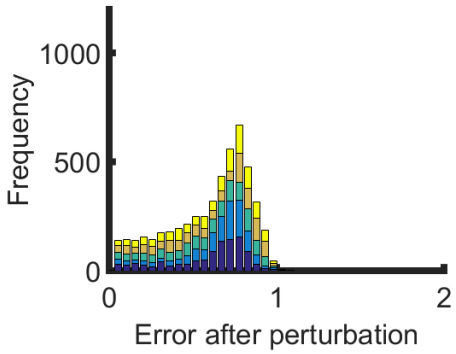

Figure 5: Distribution of errors after a global perturbation with standard FORCE. Five networks (represented by each color) with architecture A were generated and tested 1,000 times with a different global perturbation where all the non-zero synaptic weights were disrupted with a Gaussian value equal to 0.1% of their total value. This manipulation leads to a total change equivalent to removing 1 neurons.

the reservoir's synapses. A perturbation vector was obtained by multiplying the values of all non-zero synapses with a given proportion of disruption. This vector was then shuffled and added to the values of the non-zero synapses. After this manipulation, the performance was a lot less degraded than when comparable number of synapses or neurons were removed (Fig. 4(g-i)). The Δw (i.e., the total quantity of change in weights between the original and the perturbed network) is exactly the same for a given proportion of perturbation. For instance, when 1% of all synapses are removed or the value of all synapses are changed by 1% of their total value, the difference in total weight between the original and modified connectivity matrix (Δw) will be the same, on average. This greater structural robustness for global perturbations was also observed with standard FORCE (Fig. 5). This outlines an important difference between the two types of perturbation that is not captured by the mean amount of change but by the way it is distributed. So far we designed perturbations on networks with a size of 1,000 neurons to match most of the results from (Laje & Buonomano, 2013; Sussillo & Abbott, 2009). When applied to net-

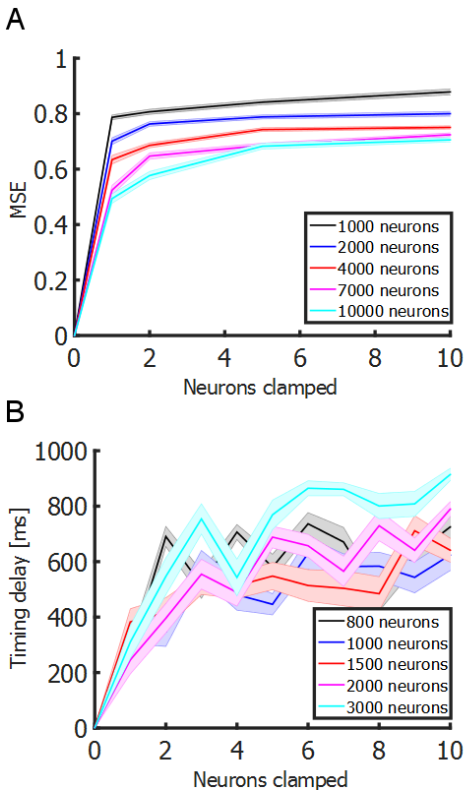

Figure 6: Clamping neurons from reservoirs of different sizes. **a** Average MSE with architecture A for reservoirs of various sizes. Each network is initially trained until the MSE is below 0.01. Then, different proportions of neurons are clamped to zero. **b** Average delay in timing for reservoirs of various sizes.

works of different sizes, perturbations were affected by the absolute number (rather than the proportion) of neurons clamped. The average MAE for different number of neurons clamped decreased slightly as the total number of neurons in the reservoir was increased with architecture A (Fig. 6). But the smallest perturbation on the largest reservoir (1 neuron clamped out of 10,000) still led to catastrophic disruptions of the network's output (average of 0.5 MAE). With architecture D, the average error of a perturbation was not related to the size of the reservoir in the testable range (larger networks are harder to train). In other words, clamping a given number of neurons had a relatively similar impact on performance regardless of the size of the networks. In summary, structural perturbations of the reservoir had a devastating impact on the task learned by the readout unit when

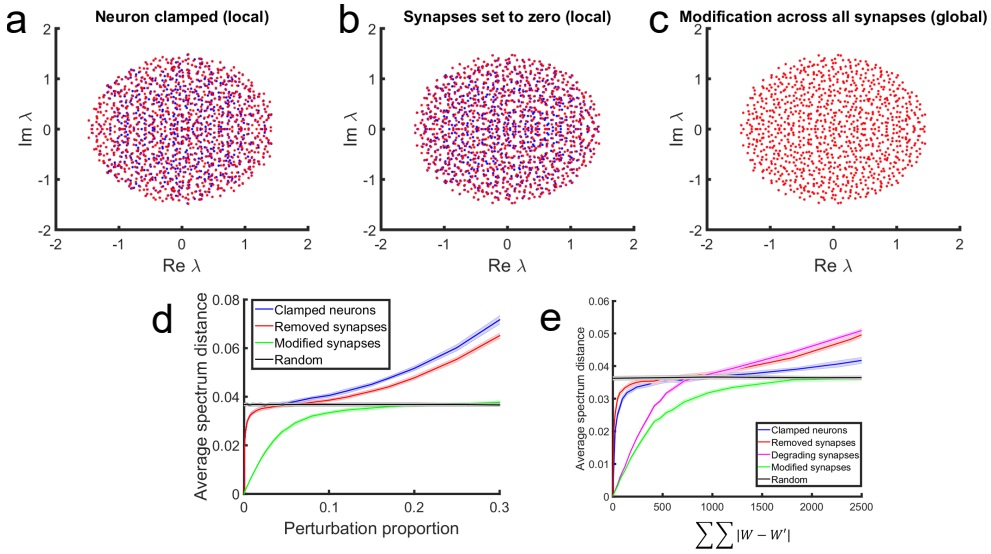

Figure 7: Eigenvalues of perturbed networks. **a** Eigenvalues of a random network (blue) and of the same network with N-1 neurons (red) (Δw = 21). **b** Eigenvalues of a reservoir (blue) and of the same reservoir where 0.1% of the synapses have been set to zero (Δw = 11.40). **c** Eigenvalues of a reservoir (blue) and of the same reservoir where all non-zero synapses have been disrupted by values following a Gaussian distribution (red) for a total disruption equal to 0.1% of their total weight (Δw = 11.87). **d** Average distance between the eigenvalues of the original and the perturbed network as the size of the perturbation increased. Each proportion represents either the number of units removed or the weights changed. The random condition is constant across all levels of perturbation and shows the distance expected between two random networks. Shaded areas represent the standard error of the mean. **e** Same as **d**, but showing the distance in eigenvalue spectrum as a function of the absolute difference between the perturbed and unperturbed connectivity matrix.

applied in a local fashion (i.e. synapses set to zero, as opposed to the global Gaussian perturbations). In the following section, we address in detail the impact of such perturbations on the network's dynamics and activity features.

2.3.3 *Structural perturbations and eigenvalues*

We computed the eigenvalues of the random networks connectivity matrix to gain some insight on the effect of the perturbations. More specifically, we aimed to characterize the contrast between the dra-

matic impact of the structural perturbations that we investigated to the milder effect of similar perturbations reported in previous studies (Laje & Buonomano, 2013; Sussillo & Barak, 2013; Sussillo et al., 2015), which used global perturbations (Gaussian perturbation to the reservoir's connectivity matrix) as opposed to the local perturbations that we introduced. We computed the eigenvalues of random connectivity matrices (N = 1,000), where each non-zero connection is drawn from a Gaussian distribution (μ = 0, σ (gain) = 1.5), and are equivalent to the connectivity matrices of untrained reservoirs. Therefore, we compared the impact of different types of structural perturbations outside of a training task. With this procedure, it is expected that the distribution of eigenvalues lies within a circle with a radius equal to the gain of the network's connections (see e.g. Bai, 1997). The deviation between the eigenvalues of the original and the damaged network (one neuron clamped) is quite large (Fig. 7(a)). This effect explains the discrepancy between the output and the target of the two conditions, as the eigenvalues govern the dynamics of the network.

To gain a better understanding of the different perturbations, we also tested the impact of removing synapses and of Gaussian perturbations to every synapse. We clamped 0.1% of the neurons (1/1,000 neurons, Fig. 7(a)), removed 0.1% of the synapses (100/100,000 synapses, Fig. 7(b)) and changed 0.1% of the synapses' weights (Fig. 7(c)). This results in the same amount of change (Δw) for the last two conditions (Δw = 11), and about double the amount of change when one neuron is clamped, because both the efferent and afferent connections are removed. In addition, clamping one neuron effectively removes one dimension from the eigenvalue spectrum (i.e., it reduces the rank of the connectivity matrix by one). When visually inspecting the eigenvalue spectrum, it is obvious that removing individual synapses and

neurons has a much greater impact than splitting the same amount of change across all synapses. We then formally compared the distance between the original and the perturbed eigenvalue spectrum. To do so, we computed the average distance between each eigenvalue of the original network and its closest eigenvalue in the perturbed network. For each level of perturbation and each type of perturbation, we generated 50 pairs of original/perturbed networks and computed their average distance (Fig. 7(d)). Different profiles emerged for the different types of perturbation. Local perturbations created large disruptions on the eigenvalue spectrum, where the distance between the original and perturbed networks reached the same value as for two independently generated random networks when about 5% of their units were removed. The curve for clamped neurons had a slightly larger distance because the number of dimensions decreased, whereas it remained constant when a proportion of synapses were removed (until extreme values are reached). This contrasts with the impact of the change in synaptic weights, where between 15% and 20% of their value had to be changed to reach the same distance as would be expected between two random networks. It is interesting to note that even if the Δw is almost doubled when the same proportion of neurons as synapses was removed, their impact on the spectrum distance was similar until it reached the average distance between randomly generated networks, which is concordant with their similar impact on the timing task with architecture D (Fig. 3(e) and Fig. 4(a)). As neurons and synapses were removed, the radius of the eigenvalue spectrum shrank, which explains the increasing eigenvalue distances past the random condition. Because we added the equivalent of Gaussian perturbations with a mean of zero on the network's synapses, the radius of the eigenvalue spectrum remained the same. In Fig. 7(e),

we show the distance in eigenvalue spectrum as a function of absolute difference in the perturbed and unperturbed connectivity matrix. We also examined the impact of an additional type of perturbations termed "synaptic degradation", where each synapse of the network received a random perturbation of the opposite sign so that they would gradually shrink to 0. This type of perturbation is more comparable to removing neurons and synapses because it reduces the rank of the connectivity matrix as the perturbation proportion increases. Difference in eigenspectra should manifest in differences of network trajectories. With innate learning, the network is able to return to the trained trajectory after the injection of a substantial pulse in the reservoir (Laje & Buonomano, 2013). The effect of this form of perturbation might be akin to distributing the structural perturbation across the whole reservoir (Fig. 4(g-i)). Based on the performance of the network on the trained task, the reservoir was no longer able to return to the original trajectory when a structural perturbation targets a specific portion of the network. Further analysis shows that despite the difference between the activity of the original and perturbed reservoirs, the trajectories of damaged networks appeared to be stable, because the new trajectory was still repeatable across trials (Fig. 8). Therefore, structural perturbations didn't increase the chaos and noise of the network, but rather changed its intrinsic dynamics in a way that previously learned tasks couldn't be performed. In summary, the eigenvalues analysis confirms the large impact of local structural perturbations on the network independently of the training algorithms.

2.3.4 *Impact of network connectivity*

We explored alternatives to the standard patterns of connectivity in reservoirs that could improve the tolerance of these networks to struc-

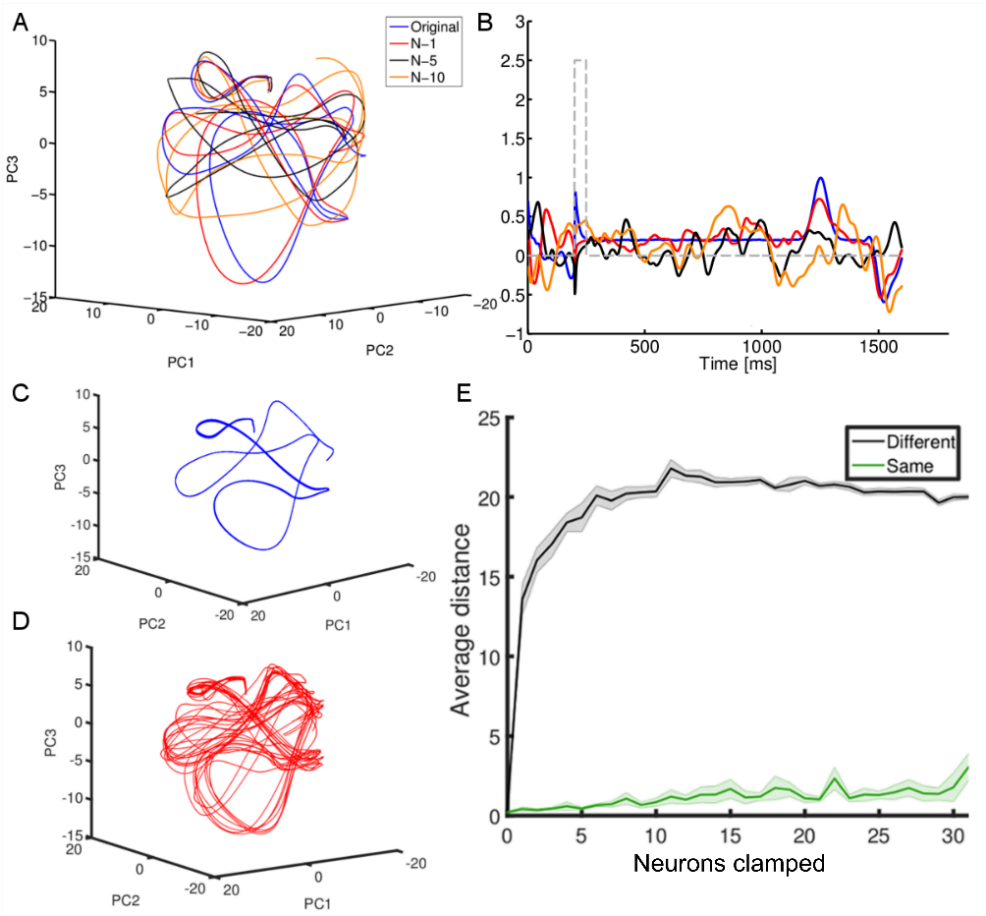

Figure 8: Impact of structural perturbations on networks trajectories. **a** Projection on the three principal components of the trajectories of reservoirs of size of N, N-1, N-5 and N-10. **b** Outputs of the corresponding trajectories in **a** given by the readout unit. **c,d** Trajectories of networks with N and N-1 neurons following the offset of the input signal on ten trials with different initial conditions. **e** Average distance (Euclidean distance) between the trajectories of 20 trials with 10 networks of size N-n for which n is the same subset on every trial (green) or a different subset for each trial (black). The Euclidean distance between each trajectory was computed between a reference trajectory (chosen randomly for each condition) and the other perturbed trajectories. We created 10 networks from which we generated 20 + 1 (for the reference trajectory) perturbed trajectories for each level of perturbation. For every time steps after the end of the input pulse, the Euclidean distance was computed between the reference and the other perturbed trajectories until the end of the target function. The distance at each time step was then averaged for each trajectory, and then for each network across all levels of damage.

tural perturbations. In previous simulations with architecture A, we used a sparse connectivity where only 10% of all possible connections were set to non-zero values (the standard value for most reservoirs models). However, the total number of connections that each neuron makes to its counterparts might have an impact on their importance on the overall network dynamics. We tested a lower range of sparseness that remained biologically relevant (1% to 10%), and found no correlation between sparseness and resistance of the networks to structural perturbations (r = -0.02, p = n.s.). Next, we looked into different patterns of connectivity as potential solutions to the sensitivity of the model to structural perturbations. In typical reservoir models, the number of connections of each neuron follows a normal distribution and each neuron has the same probability to connect to other neurons in the reservoirs. However, several studies suggest that the number of synapses per neuron in the cortex might follow either a power-law (slope of -1.3) (Bonifazi et al., 2009) or exponential distribution (Perin, Berger, & Markram, 2011). This type of connectivity has been linked to higher resilience of networks to random failure (Rubinov & Sporns, 2010). We tested with architecture A whether these types of connectivity could help improve the resistance of reservoirs to structural perturbations. We applied these distributions to the afferent and the efferent connections separately. This means that when one of these distributions was used for either the afferent or efferent connections, the other type was kept normally distributed.

Our results show that exponential and power-law (Fig. 9(a), left and middle) distributions slightly improve the resistance of the networks (larger left tail of the distributions compared to Fig. 2(d)), but only when applied to afferent connections. In this configuration, most neurons received a very small number of connections from other neu-

rons whereas a few neurons received a high number of connections. However, the MAE of most trials was still high after clamping a single neuron.

2.3.5 *Network modularity*

There is a body of work that shows that the brain is modular on different scales (Meunier, Lambiotte, & Bullmore, 2010). On the meso scale, the cortex is arranged in a multitude of cortical columns (Buxhoeveden & Casanova, 2002; Mountcastle, 1997). On a higher level, the cerebral cortex has been shown to be organized in different sub-regions that have non-symmetrical ways to communicate with each other (Keller et al., 2014). Modular connectivity has been associated with neurobiologically realistic regimes of activity in simulated networks (Rubinov, Sporns, Thivierge, & Breakspear, 2011). Additionally, clustered and modular reservoirs have been linked to increased stability in echo-state networks (Jarvis, Rotter, & Egert, 2010; Li, Zhong, Xue, & Zhang, 2015). If cortical circuits were to work accordingly to reservoir computing principles, the reservoirs might have to be distributed across different sub-regions, or compartmentalized within each region to avoid the propagation of the error caused by the failure of one of its components. We started by making networks made of modules without connections between them, then we progressively added inter-module connections to assess the impact of modularity under more realistic conditions. Fig. 9(b) shows a toy model of a reservoir made of several modules that were initialized as smaller random networks. When a neuron was clamped in one module, only this module was affected by the perturbation and the other ones remained intact (Fig. 9(c)). Traces in blue show the trajectories (on three different trials) of two modules that were not directly affected by a

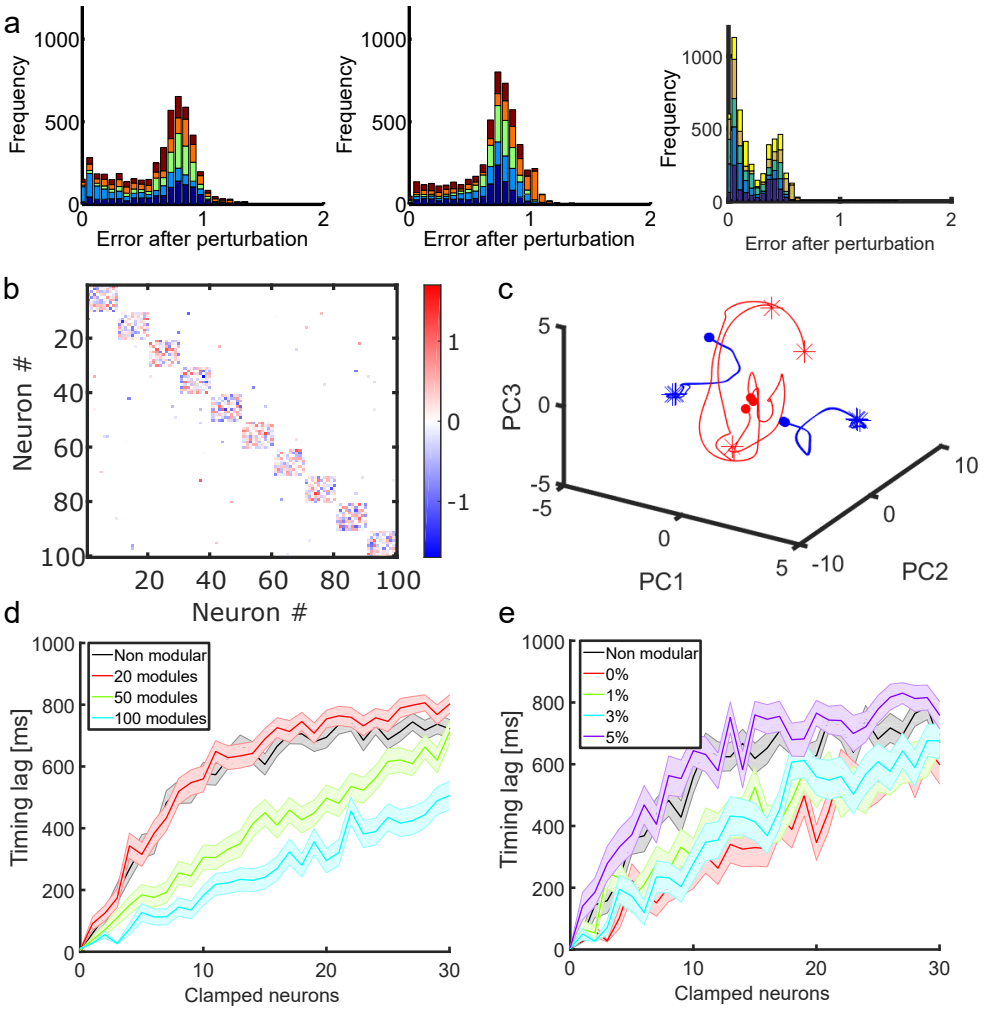

Figure 9: Error distribution of networks with an exponential, power-law and modular connectivity. **a** All distributions were obtained from architecture A (5,000 trials each). Each color represents the MAE distribution from a different network, from which each neuron was clamped once. Error distribution of networks (N-1) with an exponential distribution (left) of afferent connections and a power-law distribution with a slope of -1.3 (middle) of afferent connections, and (right) a reservoir of 2,000 neurons with 100 modules (of 20 neurons each). **b** Connectivity matrix of a toy example for a network with 10 modules of 10 neurons each. For our analysis, the size (s) and number of modules (n) were the following (s by n): 1,000 by 1, 50 by 20, 20 by 50 and 10 by 100. Modules were originally fully connected without self-connections (diagonals set at zero). Then, 5% of these connections were rewired to connect modules to each other. **c** Toy example of the trajectories of different modules of a reservoir (100 by 10, probability of connectivity within modules = 1 (without self-connections), probability of connectivity between modules = 0.03). Red traces show the trajectories of the perturbed module (circle = start, star = end) for 300 ms. Blue traces show the trajectories of two modules that are not directly perturbed. **d** Timing lag of the output based on the time it crosses a threshold for reservoirs with different configurations of modules. **e** Timing lag of modular networks of size 10 by 100 where 0%, 1%, 3% and 5% of synapses that were originally connecting neurons within modules have been rewired between modules. Shaded areas represent the standard error of the mean.

perturbation. Traces in red show the trajectories of the module where one neuron was clamped. This illustrates the resistance of spared modules to noise in the perturbed modules, as long as the number of connections between modules remains low. When implemented with the models, there was indeed an improvement in the resistance to perturbations on the complex sinusoidal task with architecture A (Fig. 9(a), right). There is also an improvement with architecture D and the timing task, where Fig. 9(d) illustrates the impact of perturbations on different module size (reservoir sizes are kept constant). Of course, performance degraded as we added inter-module connections. Fig. 9(e) shows the impact of adding connections between modules, where 0%, 1%, 3% and 5% of the synapses within each module were randomly rewired to neurons from different modules. While modularity reduces the vulnerability, it does not eliminate the problem and requires the network to implement strict parameters values to improve resilience.

2.3.6 *Driving reservoirs with oscillations*

The above findings show that recurrent networks operating in the chaotic regime are extremely sensitive to local structural perturbations. Based on these results, we conclude that it is unlikely that recurrent networks of firing rate neurons can produce reliable trajectories as autonomous systems (i.e., in absence of ongoing external drives) with the reservoir models tested in this study. This is in accordance with results from studies with recurrent networks of spiking neurons (Banerjee et al., 2007; London et al., 2010). In this section, we describe an alternative autonomous model where the reservoir is non-autonomous but is driven by an autonomous input layer. For the duration of a trial, this input layer is continuously stimulating the reservoir

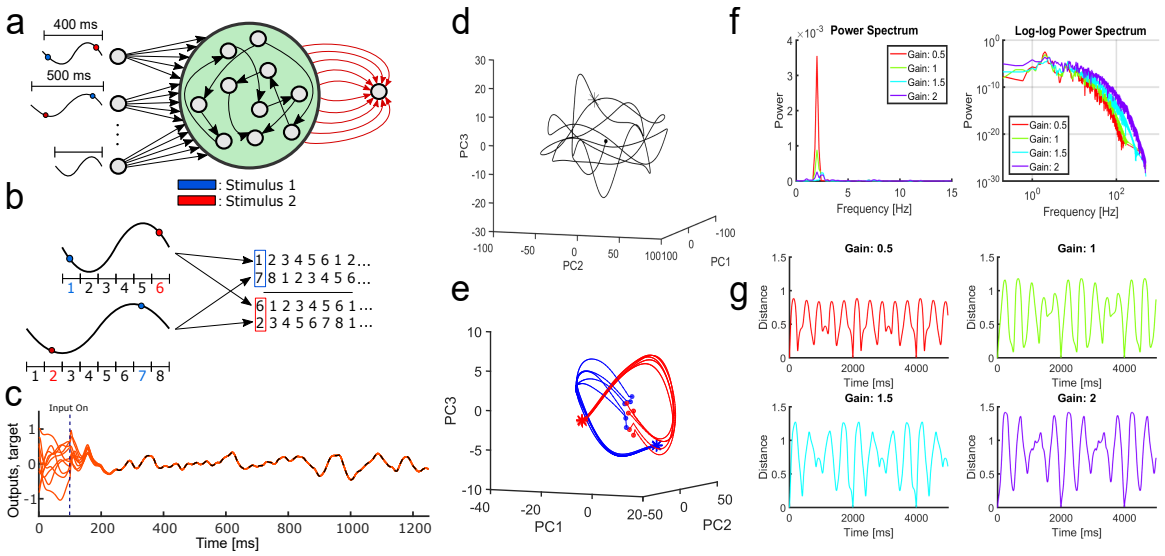

Figure 10: Generation of stable trajectories with oscillators. **a** The activity of the oscillators is controlling the dynamics of the reservoir. Multiple oscillators can be used to increase the length of the trajectory they can generate. The first two oscillators in this example can create a trajectory that will not repeat itself until after 2,000 ms (least common multiple of 400 ms and 500 ms) of ongoing stimulation. The stimuli (e.g., a go cue) can be seen as triggers that align the phases of the oscillators in the same configuration on every trial. **b** The phase configuration can be created by an input that aligns the oscillations at some given phases corresponding to the segment of the trajectory that is required. **c** Outputs of the driven network in **a** that was trained on a randomly generated time-varying function with frequency components below 15 Hz.**d** Projection of the reservoir's activity on the principal components when driven by the first two oscillators. The trajectory repeat itself after 2,000 ms (green arrow). The initial segment before the green arrow represents the convergence of the reservoir's state to the dynamic attractor created by the input. **e** Five trials with random initial conditions (200 ms) created by the two stimuli. The filled circles represent the initial states and the stars mark the end of the 200 ms stimulation. **f** Left: power spectrum of the network activity. Right: power spectrum in log-log coordinates. **g** Distance between a random time step along the trajectory and all subsequent steps (average for all N neurons of the reservoir) for the different network gains. The trajectory is repeated every 2,000 ms, and the average distance between each point on the trajectory is increased with the network gain.

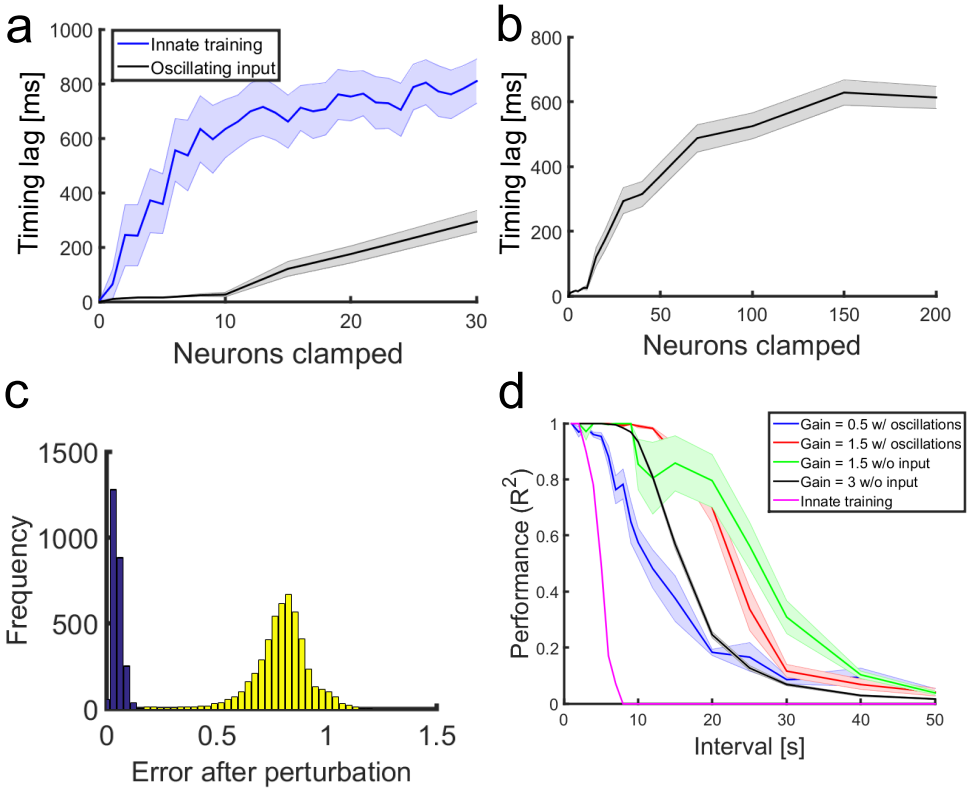

Figure 11: Performance of the driven model. **a** Timing lag between the output and the target after clamping different numbers of neurons from the reservoir. **b** Timing lag for the driven model after greater proportions of neurons have been removed from the reservoir. The network reaches random performance (500 ms lag) after about 75 neurons out of 1,000 are clamped, compared to 8 neurons for innate training. **c** Error on the complex sinusoidal task after clamping a single neuron of the new model (blue, average MAE of 0.05) and architecture A (yellow, average MAE of 0.79). The frequency of the oscillators was set manually to match the length of a full cycle of the complex sinusoid (1.2 seconds). **d** Performance (R^2) on the timing task for different timing interval lengths. Blue/Red: the network is driven by multiple oscillators, with reservoir gains of 0.5 and 1.5 respectively. A higher gain increases the network performance. Green/Black: performance of architecture D with no recurrent training where the trajectory of the reservoir is repeated by resetting the initial state of the network to the same values on each trial. Magenta: performance of innate training. For **a**,**b** and **d**, we generated 10 networks (per conditions for **d**) where the frequencies of the oscillators were drawn for a uniform distribution with a minimum of 1 Hz and a maximum of 5 Hz. For **a** and **b**, we clamped 10 subsets of neurons per network, for a total of 100 tests per data point. Shaded areas represent the standard error of the mean.

with an ongoing periodic drive where each input unit behave as an independent oscillator. Such oscillatory dynamics can be endogenously generated by small neural circuits with precise connectivity between their excitatory and inhibitory components (Yuste, MacLean, Smith, & Lansner, 2005). This framework allows the production of stable neural trajectories that are the result of the interaction between the periodic input, the state and the structure of the reservoir. This model is resilient to local and global perturbations and conserves the computational properties offered by chaotic neural networks (Bertschinger & Natschläger, 2004). Previous work showed that chaotic networks can be stabilized with an oscillatory input (Rajan et al., 2010). However, an important limitation of this approach is that the duration of the resulting neural pattern can't exceed the period of the oscillation. Alternatively, in our model, combining a population of oscillatory units provides a major advantage: it increases the duration of the pattern that the input layer can produce. The mechanisms underlying this effect can be explained by the residue number system (Soderstrand, Jenkins, Jullien, & Taylor, 1986). In short, oscillators of different periods can repeat their combined initial state after a delay equal to the least common multiple of all their combined periods, as long as the ratio of their periods is a rational number. This creates a powerful mechanism by which the network creates meaningful and robust neural trajectories. Fig. 10(a) illustrates the general mechanisms of this model. Each input unit oscillates at a constant frequency throughout the duration of the stimulation. The combination of oscillators with different periods (that are not multiple of each other) yields a much longer input period than the period of the slowest oscillator. For instance, the least common multiple of the two oscillators on Fig. 10(a) is 2,000 ms, which means the trajectory will repeat itself every two

seconds. It is well-known mathematically that the combined activity of multiple oscillators can create very rich dynamics. To visualize this, consider two linear oscillators whose phase evolution are given by $\theta_1(t) = w_1 t(\%2\pi)$ and $\theta_2(t) = w_2(\%2\pi)$ respectively, where θ represents an angle on the unit circle $[0, 2\pi]$. The combined state of the pair of oscillators, $(\theta_1(t), \theta_2(t))$ is a two-dimensional coordinate on the Cartesian product of two unit circles, or in other terms, a 2-Torus. Suppose, without loss of generality that the pair of oscillators both start at $\theta = 0$. The question is: will their combined trajectory ever wind back to (0,0)? If so, how many cycles of each oscillator will it take? For both oscillators to come back to the origin at the same time means that there exists a single time t such that $w_1 t$ and $w_2 t$ are both exact multiples of 2π. Suppose then that m and n are integers such that $w_1 t = 2\pi m$ and $w_2 t = 2\pi n$, then it follows that $w_2/w_1 = n/m$, which is only possible if the fraction $w2/w1$ is a rational number. If this is the case, then the smallest valid integers m and n correspond to the respective number of cycles of each oscillator that combine to create a periodic trajectory on the torus. If the ratio of frequencies is not rational, it can be shown that the trajectory will never repeat itself and that is will densely fill the entire torus over time (see e.g. Brin & Stuck, 2002). It is easy to see how this mechanism generalizes to the combination of multiple oscillators. The rich behaviour that emerges from the joint dynamics of uncoupled oscillators is a powerful mechanism that can be used in a variety of contexts (Buhusi & Meck, 2005; Miall, 1989). The injection of such periodic activity in reservoirs requires no training of their recurrent connections, and allows the production of an unlimited number of trajectories where their total length is dependent on the input and not on the network (Fig. 10(a,b)). This is a major improvement compared to the innate training where only

innate trajectories of the network could be stabilized, and those trajectories were greatly limited in duration (about 8 seconds) and quantity. The weights of the connections of the oscillators were drawn from a Gaussian distribution and were sparsely connected to the reservoir. Fig. 10(c) shows the output of a readout trained to produce a randomly generated time-varying output (white noise low pass filtered at 15 hz). Each trace shows a given trial with random initial conditions where the oscillators are turned on after 100 ms (blue dashed line) and the target output starts at 250 ms (black dashed line). Fig. 10(d) shows one complete trajectory, based on the first two oscillating units of Fig. 10(a). After a transient segment where the network converges from its initial state to the dynamic attractor (green arrow) created by the injection of the oscillating input in the reservoir, the neural trajectory will repeat itself every two seconds as long as the oscillators are active and keep their relative phase shift. Some segments of this full trajectory can be evoked by stimuli that either reset or activate the oscillators at given phases (Fig. 10(b)), or that turn them on sequentially in order to match some predetermined phase shifts. Every time the oscillators are initialized at some predetermined phases, they can be used to evoke segments of the full trajectories (Fig. 10(e)), that can then be decoded by a readout unit. The projection of the oscillators' activity in a recurrent network serves two principal roles. First, it creates a high-dimensional representation of the low-dimensional input (Häusler et al., 2003). Secondly, it takes advantage of the recurrent connections of the network to decorrelate the highly correlated inputs that each neuron receives. Fig. 10(e) shows that as the network gain was increased, the peak in the power spectrum (2 Hz and 2.5 Hz based on the example of Fig. 10(a)) decreased, while the power of the other frequencies increased, which shows that the overall correlation

of the network decreased. Fig. 10(g) shows the distance (average for all neurons of the reservoir) between a randomly selected time step of the network's trajectory and all subsequent time steps. The distance reaches 0 every 2,000 ms for all conditions, which shows that the network is back to its initial state. However, as the network gain increased, the overall distance between this point and the other states of the network increased, which suggests better discriminability between each step of the trajectory. The resilience of the driven model was tested on the previous tasks (timing and complex sinusoid), although the activity of the reservoir driven with oscillations could be used with many other tasks where the output is a time-varying function. This model displayed considerably improved resistance to the structural perturbations that were completely shutting down the ability of the previous models to perform their task. Fig. 11(a,b) shows the performance after different numbers of neurons were clamped from the reservoir of the driven model and from reservoirs with innate training. With innate training, networks reached random performance when less than 10 neurons were clamped, whereas the driven model could sustain the loss of about 75 neurons before it reached the same disruption in performance. Fig. 11(c) shows the performance on 2,500 trials on the sinusoid production with the driven model where the period of the input oscillators matches the period of the target sinusoidal wave and 5,000 trials with architecture A (same as Fig. 2(d)). There is a large improvement with the new model where almost all networks were still successful after the perturbation compared to architecture A where no networks were able to sustain the loss of a single neuron. The maximum length of the timing interval is significantly greater with the driven model than with innate training. To get a reliable estimate of the capacity of the new model, we generated the

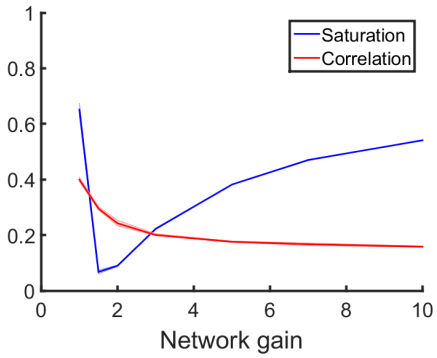

Figure 12: Saturation and correlation of chaotic networks with different gain values. Saturation and correlation (defined below) of networks (N = 1000). There is a trade-off between the optimal saturation and network correlation value.

performance curve of different interval lengths with the spontaneous activity of reservoirs that was replayed multiple times with the exact same initial conditions and without noise. The performance of the driven model was close to the optimal performance obtained when using the spontaneous activity of chaotic networks (Fig. 11(d)). The lower performance of the spontaneous activity of the network with a gain of 3 was probably caused by the saturation of the reservoir's neurons (Fig. 12). This performance also appears to be dependent on the strength of the reservoir's connections for the driven model, where stronger connections (gain = 1.5) lead to better performances than lower gain values. There was a trade-off between the stability of the trajectory, the saturation and the correlation of the network's units (Fig. 13). In summary, we showed that a weak periodic drive both helps a network to remain robust to structural perturbations and increase its performance compared to previous implementations.

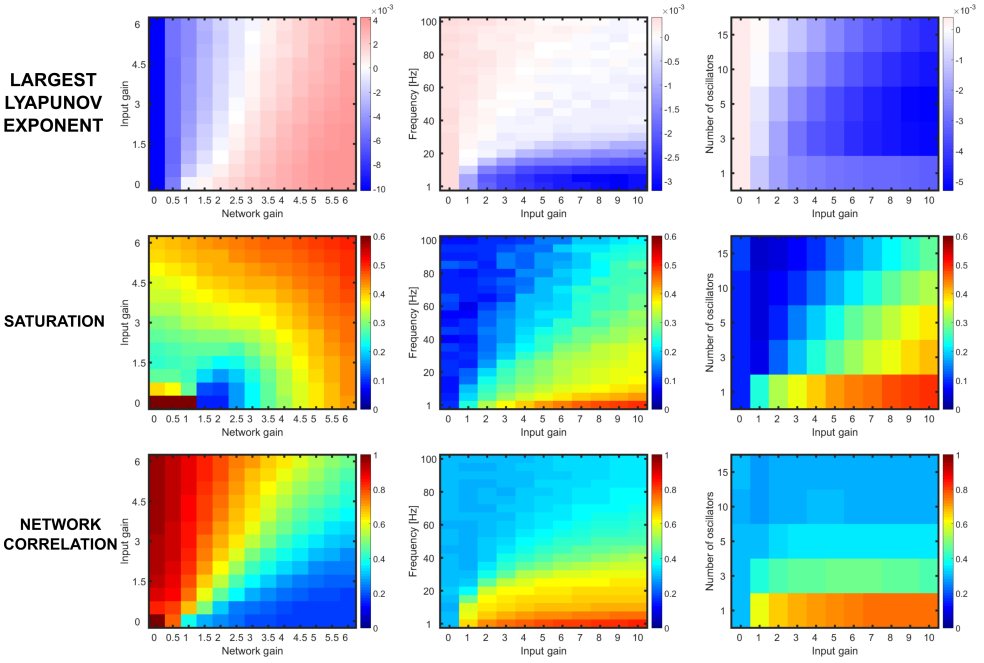

Figure 13: Impact of the network parameters on its dynamics. Effect of network gain, input gain, input frequency and number of oscillators on the Largest Lyapunov Exponent (LLE), network saturation and network correlation. Top row: LLE, when LLE > 0 the network is chaotic (red) and when LLE <0, the network is stable (blue) and will return to the dynamic attractor after a perturbation. Middle row: network saturation, 0 = lowest saturation value (blue), 1 = highest saturation value (red). Bottom row: network correlation, 0 = lowest correlation value (blue), 1 = highest correlation value (red).

2.4 Discussion

2.4.1 *Summary of results*

Our results show that reservoir models that were trained to produce precise patterns of activity were highly vulnerable to small structural perturbations. With every architecture that we tested, clamping a single neuron, or setting the weights of a very small subset of synapses to zero, had either a strong impact or completely disrupted the networks' performance on the selected task. This is at odds with the apparent resilience of the brain to minor functional and structural perturbations. For instance, the cerebellum shows considerable resistance to lesions on tasks related to interval timing (Perrett et al., 1993). The impact of these lesions on the latency of behavioural responses can be absent. In contrast, when the same amount of perturbation was distributed across all connections, the performance degradation was milder. We found little or no improvement in resistance to perturbations when increasing the dimensionality of the reservoirs. Thus, in the range of reservoir sizes that we tested, the absolute number of disrupted neurons, and not the proportion of neurons clamped, explained the impact on performance. Changing the distribution of connections in the reservoir had little benefit in terms of resistance, except that a small gain was obtained with an exponential and a power-law distribution of afferent connections. We found a marked improvement in resistance with modular networks, but even in this scenario networks were still vulnerable to local structural perturbations, and a very sparse connectivity between modules was required to benefit from this type of connectivity. Finally, we introduced an alternative model, where the reservoir was driven by a layer of oscillators. This architecture could learn longer timing intervals, was simpler to train

and was more resilient to local perturbations than other implementations. It is known that reservoir networks driven with inputs are more stable than non-driven network(Sussillo & Abbott, 2009), and in this work, we showed that driving reservoirs also generate structural robustness. In models that are not driven by external inputs, neurons of the reservoir are only influenced by each other's activity. Conversely, in models that are driven by external inputs, neurons are also influenced by a stable source of external activation that is preserved. This external signal will therefore help to counteract the error created by the perturbation in the network. In other words, the external drive helps to guide the network to driven trajectories, which increase robustness to changes in connectivity. The extreme sensitivity of the reservoir models that we tested might not be an issue in many tasks where a readout unit is used to discriminate between a few states of the reservoir. This is usually the case for classification tasks where as long as the resulting trajectory remains in the same subspace (e.g. basin of attraction of a fixed point attractor), the readout unit will retain its ability to generate the appropriate output. When tested on a 3-Bit Flip-Flop task (Sussillo & Barak, 2013), the networks retained a large part of their ability to perform the task (Fig. 14), compared to when the same architecture was trained to reproduce a complex sinusoid. Issues arise when the readout unit is trained to produce complex output functions based on precise temporal activity in the reservoir. As shown in our results, very small perturbations to the structure of a network cause a very large impact on the network's dynamics, leading to very different trajectories.

This has an important impact on computations since the main source of disruption in the network's performance comes from the deviation of the reservoir's trajectory. This is highlighted by the fact that the

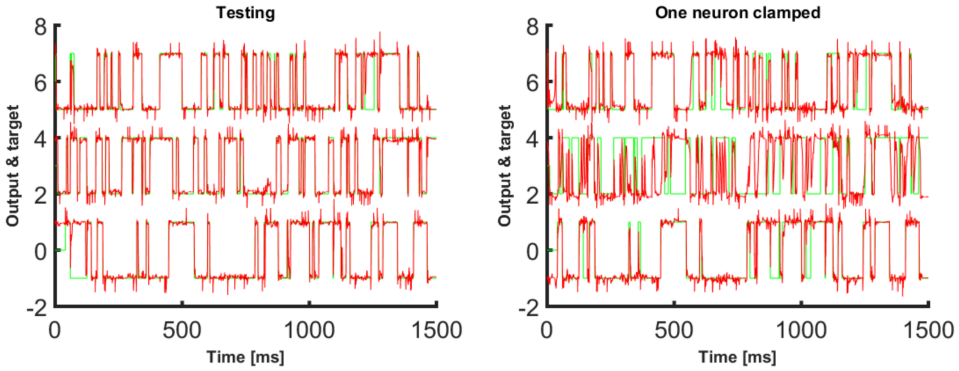

Figure 14: Perturbation on the three bit flip-flop task. Performance of the network of architecture A, with an additional layer of input and three readout units, implementation from (Jang, 2015) adapted from (Sussillo & Barak, 2013), after a single neuron out of 1,000 was clamped. The performance of the network is largely preserved after clamping a single neuron, MSE after training = 0.14, MSE after perturbation = 0.45.

networks can sustain a relatively large loss of synapses connecting the reservoir to the readout unit before being significantly impaired. Therefore, it is unlikely that the main factor explaining the impaired performance is the loss of dimensions to the solution found by the readout. One mechanism that hasn't been extensively tested in the current work is the ability of a network to modify its synapses to compensate for perturbations. The training algorithms that we used are applied to every synapse of either the reservoir or the readout. Therefore, retraining the network after such perturbations generates a completely new solution equivalent to training a new network. A localized re-training (e.g., only a subset of neurons/synapses can be modified) could be used to compensate for synaptic failure in order to retain the configuration of the previous network. When only a subset of connections of the readout units was retrained, the performance was recovered when the size of this subset reached about 10% of all synapses (randomly selected) (Fig. 15). However, in the case of multiple readouts connecting to the reservoir, the total number of synapses to retrain would increase rapidly. Therefore, retraining the reservoir

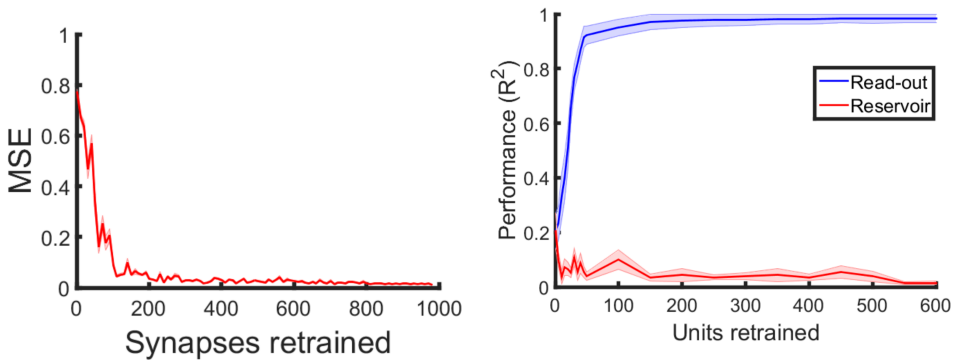

Figure 15: Retraining networks after perturbations. Left: performance of networks with a reservoir size of 1,000 neurons with one neuron clamped after different random proportions of synapses on the readout units were allowed to be retrained for the same duration and number of iterations as the original training. About 200 synapses were required for the network to recover the loss of performance caused by the perturbations. Right: Performance of networks with a reservoir size of 1,000 neurons with 5 neurons clamped after different proportions of neurons (or synapses in the case of the readout) are retrained. Retraining the neurons in the reservoir didn't increase the performance of the network, possibly because the algorithm can only learn innate trajectories. Clamping a few neurons modifies the innate trajectory and the original trajectory is no longer in the natural repertoire of the network, which might explain the inability of the network to be retrained on the original trajectory. When the readout was retrained on the activity generated by the perturbed reservoir, it is able to regain most of its original performance after a few synapses (10%) have been readjusted.

would appear to be more efficient, but this option is only available to architectures that allow reservoir plasticity. However, in the case of innate training, it was almost impossible to retrain the reservoir after a perturbation, possibly because it created different innate trajectories.

The division of reservoirs into compartmentalized pools of recurrent neurons appears to be an effective mechanism to improve their resistance to structural perturbations. Without connections between the modules, the error caused by a structural perturbation is dependent on the number of modules directly affected by the perturbation. Our results showed that modular networks were more resistant to

structural perturbations than comparable random networks, due to their ability to restrain the perturbations in a cluster of neurons of the network. However, even if this configuration is beneficial for the network's resilience, this is likely not sufficient, due to the high number of modules required and the low tolerance to inter-module connections. The low number of inter-module connections required to increase to resilience isn't biologically plausible (Meunier et al., 2010).

2.4.2 *Biological relevance of the driven model*

The driven model presented here relies on the injection of oscillations into neurons of the reservoir. Neuronal oscillations constitute an ubiquitous form of activity in various brain circuits, and transient increase in specific frequency bands is linked to many behavioural and cognitive outcomes (Churchland et al., 2012; Kahana, 2006; MacKay, 2004). The dynamics of the oscillators used in our model are analogous to central pattern generators (CPGs) in the brain (Hooper, 2001). These circuits have been shown to produce stable and repetitive activity in the absence of sustained sensory or central input. CPGs have mostly been used to explain stereotyped and repetitive motions and behaviours, such as breathing, walking or scratching (Hooper, 2001). Although they are usually associated with lower level brain areas such as in the brainstem and spinal cord, the cerebral cortex could also implement similar dynamics (Yuste et al., 2005). Additionally, the phase control of ongoing oscillations has been extensively studied, where reliable and systematic mechanisms for phase resetting have been shown in vitro with single cells (Farries & Wilson, 2012), in populations of neurons (Akam, Oren, Mantoan, Ferenczi, & Kullmann, 2012) and in model cells (Ermentrout, 1996). Therefore, based on the extensive reports of CPGs and phase-resetting mechanisms in

the brain, we suggest that the neural mechanisms to start, sustain and end the dynamics of the oscillators proposed in our model are biologically plausible. However, we make no claims about the nature or the implementation of such mechanisms and leave this part for further studies.

2.4.3 *Related models*

The use of oscillators with different periods as a source of neural activity underlying interval timing has been proposed in the past. The model proposed in (Miall, 1989) relies on similar principles, where the maximum interval that can be encoded by a neural network is given by the least common multiple of multi-periodic oscillators. In this model, an output unit is trained to detect the coincident activation of some pacemaker cells. These mechanisms have been adapted in the striatal beat-frequency theory (Buhusi & Meck, 2005; Lustig, Matell, & Meck, 2003), where the oscillators have origins distributed in the cortex, and where the basal ganglia act as a coincidence detector. However, these models are specific to interval timing tasks, and aren't applicable to other tasks that require a temporal component in their activity (Mauk & Buonomano, 2004). On the other hand, (Fiete, Burak, & Brookings, 2008) proposed a coding system based on similar principles for spatial navigation. In this theory, the different lengths of the lattices created by the grid cells are read by the place cells of the hippocampus based on a modular code, similar to the residue number system (Soderstrand et al., 1986). The model that we propose is exploiting both the spatial and the temporal component of the modular code, which makes it more general and powerful. Most reservoir models use supervised learning (Barak et al., 2013; Laje & Buonomano, 2013; Maass et al., 2002; Sussillo & Abbott, 2009), which

has limited biological plausibility because it implies that the target function is accessible to the system. While we employed this learning algorithm in our model, some unsupervised learning rules have been successfully applied to similar models (Hoerzer et al., 2014) and could potentially be used with our driven model.

2.4.4 *Conclusion*

In this work, we raised an important problem with the production of reliable patterns of activity by recurrent neural networks of firing rate units, which had already been highlighted in the past in spiking networks (Banerjee et al., 2007; London et al., 2010). Our study showed that chaotic neural networks that are resistant to noisy inputs and state-space perturbations (Laje & Buonomano, 2013) are no longer resilient under minimal structural perturbations. A priori, chaos and structural perturbations are not necessarily related, and our work showed that it is indeed an important distinction to make with neural circuits. The spontaneous activity of autonomous chaotic networks is a candidate mechanism to explain the brain dynamics required to execute intrinsically generated temporal processes. However, our work showed that this approach is too sensitive to structural perturbations. On the other hand, we showed that driving reservoirs with oscillating inputs is an attractive solution to generate activity that is resilient to structural perturbations. Future directions would involve the use of reservoirs made of spiking units to investigate their resilience to structural perturbations, as well as the possibility to generate useful spike patterns with oscillating inputs. Experimental studies could also be conducted to unravel the presence of trial-to-trial phase coherence of oscillatory activity related to the generation of spatiotemporal patterns.

In chapter 3, we will use a RC model with a spiking RNN. Because of its spiking activity, this model is more biologically plausible, and follows Dale's principle, has balanced excitation/inhibition, a heterogeneity of neuronal and synaptic parameters, propagation delays, and conductance-based synapses. We will investigate the potential of the multi-periodic code that we developed in chapter 2 to understand if it is plausible that cortical circuits might use similar mechanisms to learn complex temporal functions.

3 CHAPTER 3: LEARNING LONG TEMPORAL SEQUENCES IN SPIKING NETWORKS BY MULTIPLEXING NEURAL OSCILLATIONS.

Abstract

Many cognitive and behavioural tasks - such as interval timing, spatial navigation, motor control and speech - require the execution of precisely timed sequences of neural activation that cannot be fully explained by a succession of external stimuli. We use a reservoir computing framework to explain how such neural sequences can be generated and employed in temporal tasks. We propose a general solution for recurrent neural networks to autonomously produce rich patterns of activity by providing a multi-periodic oscillatory signal as input. We show that the model accurately learns a variety of tasks, including speech generation, motor control and spatial navigation. Further, the model performs temporal rescaling of natural spoken words and exhibits sequential neural activity commonly found in experimental data involving temporal processing. In the context of spatial navigation, the model learns and replays compressed sequences of place cells and captures features of neural activity such as the emergence of ripples and theta phase precession. Together, our findings suggest that combining oscillatory neuronal inputs with different frequencies provides a key mechanism to generate precisely timed sequences of activity in recurrent circuits of the brain.

3.1 Introduction

Virtually every aspect of sensory, cognitive and motor processing in biological organisms involves operations unfolding in time (Buonomano & Maass, 2009). In the brain, neuronal circuits must represent time on a variety of scales, from milliseconds to minutes and longer circadian rhythms (Buhusi & Meck, 2005). Despite increasingly sophisticated models of brain activity, time representation remains a challenging problem in computational modelling (Grondin, 2010; Paton & Buonomano, 2018). Recurrent neural networks offer a promising avenue to detect and produce precisely timed sequences of activity (Abbott et al., 2016). However, it is challenging to train these networks due to their complexity (Pascanu et al., 2012), particularly when operating in a chaotic regime associated with biological neural networks (Abarbanel et al., 2008; C. van Vreeswijk & Sompolinsky, 1996). One avenue to address this issue is with the use of reservoir computing (RC) (Jaeger, 2002; Maass et al., 2002). Under this framework, a recurrent network (the reservoir) projects onto a readout layer whose synaptic weights are adjusted to produce a desired response. However, while RC can capture some behavioural and cognitive processes (Laje & Buonomano, 2013; Nicola & Clopath, 2017; Sussillo & Abbott, 2009), it often relies on biologically implausible mechanisms (H. F. Song, Yang, & Wang, 2016). Further, current RC implementations offer little insight to understand how the brain generates activity that does not follow a strict rhythmic pattern (Abbott et al., 2016; Buonomano & Maass, 2009). That is because RC models are either restricted to learning periodic functions, or require an aperiodic input to generate an aperiodic output, thus leaving the neural origins of aperiodic activity unresolved (Abbott et al., 2016). A solution to

this problem is to train the recurrent connections of the reservoir to stabilize innate patterns of activity (Laje & Buonomano, 2013), but this approach is more computationally expensive and is sensitive to structural perturbations (Vincent-Lamarre, Lajoie, & Thivierge, 2016). To address these limitations, we propose a biologically plausible spiking recurrent model that receives multiple independent sources of neural oscillations as input. The combination of oscillators with different periods creates a multi-periodic code that serves as a time-varying input that can largely exceed the period of any of its individual components. We show that this input can be generated endogenously by distinct subnetworks, alleviating the need to train recurrent connections of the reservoir to generate long segments of aperiodic activity. Further, no feedback from the output units to the reservoir was required during training (Maass et al., 2007; Sussillo & Abbott, 2009). Thus, mixing a set of oscillators into a reservoir network provides an efficient and neurophysiologically grounded means of controlling a recurrent circuit (Vincent-Lamarre et al., 2016). Analogous mechanisms have been hypothesized in other contexts including grid cell representations (Fiete et al., 2008) and interval timing (Matell & Meck, 2004; Miall, 1989). This paper is structured as follows. First, we describe a simplified scenario where a reservoir network that receives a collection of input oscillations learns to reproduce an arbitrary time-evolving signal. Second, we extend the model to show how oscillations can be generated intrinsically by oscillatory networks that can be either embedded or external to the reservoir. Third, we show that a network can learn several tasks in parallel by "tagging" each task to a particular phase configuration of the oscillatory inputs. Fourth, we show that the activity of the reservoir network captures temporal rescaling and selectivity, two features of neural activity reported during behavioural

tasks. Fifth, we train the model to reproduce natural speech at different speeds when cued by input oscillations. Finally, we employ a variant of the model to capture hippocampal activity during spatial navigation. Together, results highlight a novel role for neural oscillations in regulating temporal processing within recurrent networks of the brain.

3.2 Methods

3.2.1 *Integrate-and-fire networks*

Driven networks

Our network consists of leaky integrate-and-fire neurons, where $N_{res} = 1,000$ for the reservoir and $N_{osc} = 500$ for each oscillatory network, by default. 80% of these neurons are selected to be excitatory while the remaining 20% are inhibitory. The membrane potential of all neurons is given by

$$C\frac{dV}{dt} = \frac{1}{R}(E_L - V) + g_{ex}(E_{ex} - V) + g_{in}(E_{in} - V) + I_{tonic} + I_{inp} \quad (16)$$

where C and R are the membrane capacitance and resistance, E_L is the leak reversal potential, g_{ex} and g_{in} are the time-dependent excitatory and inhibitory conductances, E_{ex} and E_{in} are the excitatory and inhibitory reversal potentials, I_{tonic} is a constant current applied to all neurons and I_{inp} is a time-varying input described below. Parameters were sampled from Gaussian distributions as described in Table 3.

The excitatory and inhibitory conductances, g_{ex} and g_{in} respectively, obey the following equations:

$$\tau_{ex_i} \frac{dg_{exi}}{dt} = -g_{ex_i} + \sum_{j=1}^{N_{res}} W_{i,j} G_{ex_i} \delta(t - t^{(j)} - T^{(j)}_{delay}) \tag{17}$$

$$\tau_{in_i} \frac{dg_{ini}}{dt} = -g_{in_i} + \sum_{j=1}^{N_{res}} W_{i,j} G_{in_i} \delta(t - t^{(j)} - T^{(j)}_{delay}) \tag{18}$$

where τ_{ex} and τ_{in} are the time constants of the excitatory and inhibitory conductances, and G_{in} and G_{ex} are the change in conductance from incoming spikes to excitatory and inhibitory synapses. V_θ is the spiking threshold, $t^{(j)}$ denotes the time since the last spike of the pre-synaptic neuron j (constant for every post-synaptic neuron), after which V is set to V_{reset} for a duration equal to τ_{ref}. T_{delay} is the propagation delay of the action potential. The reservoir connectivity is defined as a N_{res} X N_{res} sparse and static connectivity matrix W with a density p_{res} (probability of having a non-zero pairwise connection). The non-zero connections are drawn from a half-normal distribution $f(0, \sigma_{res})$, where $\sigma_{res} = \frac{\gamma_{res}}{\sqrt{N_{res} * p_{res}}}$ C. v. van Vreeswijk & Sompolinsky (1998). All ODEs are solved using a forward Euler method with time-step $\Delta t = 0.05$ ms.

Each reservoir neuron is connected to each of the N_{inp} input units with probability p_{inp}. The external inputs (I_{inp}) (with index inp dropped to alleviate notation) follow:

$$I_{i,k}(t) = \frac{A}{2}(\sin(2\pi f_k t + \phi_k) + 1) * M_{i,k} \tag{19}$$

where $i = 1, ..., N_{res}$ stands for the identity of the post-synaptic unit and $k = 1, ..., N_{inp}$, where N_{inp} represents the total number of inputs. f_k is the frequency of the sine wave and the initial phase ϕ_k is drawn from a uniform distribution $U(-\pi, \pi)$. The sine wave is then trans-

formed by adding 1 and dividing by 2 to limit its range to [0,A]. The input is rescaled by the connectivity weight $M_{i,k}$ from input unit k into reservoir unit i. The full input-to-reservoir connectivity matrix M is a N_{res} X N_{inp} sparse and static matrix, with a density of p_{inp}. The non-zero connections of M are drawn from a normal distribution $\mathcal{N}(0, 1)$ and A is the amplitude of the input (30 pA by default).

	Reservoir	Input (oscillatory networks)
N	1,000	500
R	100 MΩ	100 MΩ
C	200 pF	200 pF
E_L	μ = -60 mV, σ = 1.2 mV	μ = -60 mV, σ = 0.6 mV
V_θ	μ = -50 mV, σ = 0.5 mV	μ = -50 mV, σ = 0.5 mV
V_{reset}	μ = -60 mV, σ = 1.2 mV	μ = -60 mV, σ = 0.6 mV
I_{tonic}	90 pA	90 pA
T_{delay}	μ = 1 ms, σ = 0.02 ms	μ = 1 ms, σ = 0.01 ms
τ_{ref}	μ = 2 ms, σ = 0.04 ms	μ = 2 ms, σ = 0.02 ms
G_{ex}	μ = 20 pS, σ = 0.4 pS	μ = 30 pS, σ = 0.03 pS
G_{in}	μ = 160 pS, σ = 3.2 pS	μ = 140 pS, σ = 1.4 pS
τ_{ex}	μ = 20 ms, σ = 0.4 ms	μ = 20 ms, σ = 0.2 ms
τ_{in}	μ = 20 ms, σ = 0.4 ms	μ = 80 ms, σ = 0.8 ms
E_{ex}	0 mV	0 mV
E_{in}	-80 mV	-80 mV
p_{res}	0.1	1
p_{inp}	0.3	1
γ_{res}	1	1

Table 3: Reservoir parameters.

All of the reservoir's excitatory neurons project to the readout units. Their spiking activity r is filtered by a double exponential:

$$\tau_d \frac{dr_j}{dt} = -r_j + h_j \tag{20}$$

$$\tau_r \frac{dh_j}{dt} = -h_j + \frac{1}{\tau_d} \sum_{j=1}^{N} \delta(t - t^{(j)}) \tag{21}$$

where τ_r = 6 ms is the synaptic rise time and τ_d = 60 ms is the synaptic decay time. W_{out} is initialized as a N_{out} X $N_{res}^{(ex)}$ null matrix that is modified according to the learning rule described below (see training

procedure). N_{out} is the number of readout units and $N_{res}^{(ex)}$ is the number of excitatory neurons in the reservoir.

3.2.2 *Target functions*

Unless otherwise stated, the target functions employed to train the model were generated from white noise with a normal distribution $N(0, 30)$, then low-pass filtered with a cut-off at 6 Hz. To assess network performance, we computed the Pearson correlation between the output of the network and the target function.

Training procedure

Recursive Least Square

The W_{out} weight matrix is updated with the recursive least-square algorithm (Haykin, 2002; Sussillo & Abbott, 2009), adapted to spiking networks (see Nicola & Clopath, 2017 for full implementation).

Oscillatory networks

Oscillatory networks obey the same equations as the reservoir networks. However, the I_{inp} term in Eq.16 is replaced by the following step functions:

$$I^{(ex)} = \begin{cases} 0 & \text{if } t_{stim} > t, \text{ or } t > t_{end}^{ex} \\ AW_{in} & otherwise \end{cases} \tag{22}$$

$$I^{(in)} = \begin{cases} 0 & \text{if } t_{stim} > t, \text{ or } t > t_{end}^{in} \\ AW_{in} & otherwise \end{cases} \tag{23}$$

where t_{stim} = 500 ms, t_{end}^{ex} denotes the end of the excitatory pulse and t_{end}^{in} (different across oscillatory networks) is the end of the inhibitory

pulse, where $t^{ex}_{end} > t^{in}_{end}$. W_{in} is a dense matrix with N_{osc} X N_{inp} elements representing the connections from the tonic inputs to the oscillatory networks, where N_{inp} is the number of oscillatory networks projecting to the reservoir and $A = 20$ pA.

The oscillatory networks each project to the reservoir with a N_{res} X $(N_{osc} * N_{inp})$ sparse and static connectivity matrix M, where each oscillatory network projects to the reservoir with a density of $p_{inp} = 0.5$. The non-zero connections are drawn from a normal distribution $\mathcal{N}(0, \sigma_{inp})$, where $\sigma_{inp} = \frac{\gamma_{inp}}{\sqrt{p_{inp} N_{res} N_{osc}}}$ and $\gamma_{inp} = 10$ (a.u.) by default.

Where stated in the Results, the reservoir projects back to the oscillatory networks with a $(N_{osc} * N_{inp})$ X N_{res} sparse and static connectivity matrix M', where each reservoir unit projects to the oscillatory networks with a density of $p_{fb} = 0.5$. The non-zero connections are drawn from a normal distribution $\mathcal{N}(0, \sigma_{fb})$, where $\sigma_{fb} = \frac{\gamma_{fb}}{\sqrt{p_{fb} N_{res} N_{osc}}}$.

Selection of stable networks

As we were interested in regimes where the networks would produce reliable and repeatable oscillations to be used as an input to our model, we considered networks with an inter-trial correlation coefficient (10 trials) of their mean firing-rate greater than 0.95 as stable. A wide range of parameter combinations lead to reliable oscillations, but different random initialisations of networks with the same parameters can lead to drastically different behavior, both in activity type (ansynchronous and synchronous) and inter-trial reliability.

Jitter accumulation in input phase

While a perfect sinusoidal input such as the one in equation 30 allows for well-controlled simulations, it is unrealistic from a biological standpoint. To address this issue, we added jitter to the input

phase of each input unit. This was achieved by converting the static input phase injected into unit k ϕ_k to a random walk $\phi_k(t)$. First, we discretize time into non-overlapping bins of length Δt, such that $t_n = \Delta t * n$. From there, we iteratively define $\phi(t)$ (index *k* is dropped to alleviate the notation) as:

$$\phi_{t+\Delta t} = \phi_t + \varepsilon_t \iff \phi_{n+1} = \phi_n + \varepsilon_{n+1} \tag{24}$$

with

$$\phi(0) = \phi_0 \tag{25}$$

$$\varepsilon \sim \mathcal{N}(0, \sigma_\phi^2 \Delta t) \tag{26}$$

from initial value ϕ_0, sampled from a uniform distribution as specified previously. More intuitively, $\phi(t)$ can be constructed as:

$$\begin{aligned} \phi(0) &= \phi_0 \\ \phi_1 &= \phi_0 + \varepsilon_1 \\ \phi_2 &= \phi_1 + \varepsilon_2 = \phi_0 + \varepsilon_1 \\ &... \\ \phi_N &= \phi_0 + \sum_{n=1}^{N} \varepsilon_n \end{aligned} \tag{27}$$

On average, the resulting deviation from the deterministic signal, *i.e.* $E[\phi_N - \phi_0]$, is null. On the other hand, one can calculate its variance:

$$\begin{aligned} var(\phi_N - \phi_0) &= var\left(\phi_0 + \sum_{n=1}^{N} \varepsilon_n - \phi_0\right) \\ &= var\left(\sum_{n=1}^{N} \varepsilon_n\right) \end{aligned}$$

Since all ε_n are i.i.d:

$$\begin{aligned}\sum_{n=1}^{N} var(\varepsilon_n) &= N var(\varepsilon_n) \\ &= N\sigma_\phi^2 \Delta t \\ \Leftrightarrow var(\phi_N - \phi_0) &= t\sigma_\phi^2 \end{aligned} \tag{28}$$

For ease of comparison, we can express the equivalent standard deviation in degrees (see 25):

$$\sigma_{deg} = \sqrt{t\sigma_\phi^2}\frac{180}{\pi} \tag{29}$$

Model for place cells sequence formation

We employed a balanced recurrent network similar to the ones used for all other simulations, with a few key differences. The input consisted of N_{inp} = 20 oscillators with periods ranging from 7.5 to 8.5 Hz that densely projected to the reservoir (p = 1) and follow:

$$I_{i,k}(t) = \sin(2\pi f_k t + \phi_k) + 1) * M_{i,k} \tag{30}$$

C was set to 100 pF for all neurons and γ_{res} was set to 0.5. We removed the readout unit and connections, and we selected 10 random excitatory cells (N_{place}) as place cells. Those cells had parameters identical to the other reservoir excitatory units, except:

1. We set the resting potential of those cells to the mean of E_L, to avoid higher values that could lead to high spontaneous activity (that in turn can lead to spurious learning).

2. A 600 ms sine wave at 10 Hz with an amplitude of 60 pA was injected in each of the place cells at a given time representing the animal going through its place field.

3. The connections between the input oscillators and the place cells were modified following eq.31.

We modelled the environmental input (10 Hz depolarisation of CA1 place cells) based on a representation of the animal's location that was fully dependent on time. In order to explain phase precession, our model relied on an environmental input of a slightly higher frequency than the background theta oscillation, as suggested in (Lengyel, Szatmáry, & Érdi, 2003).

The learning rule seeks to optimize the connections between the oscillating inputs and the place cells in order to make them fire whenever the right phase configuration is reached (Matell & Meck, 2004; Miall, 1989).

We used a band-pass filter between 4-12 Hz to isolate the theta rhythm in the reservoir. We then used a Hilbert transform to obtain the instantaneous phase of the resulting signal.

Learning algorithm

We developed a correlation-based learning rule (Kempter, Gerstner, & van Hemmen, 1999) inspired by the results obtained by (Bittner et al., 2017) (as described below):

$$M_{i,k}(t + \Delta t) = M_{i,k} + \alpha I_{inp_k}(t) \tag{31}$$

where $i = 1, ..., N_{place}$ and $\alpha = 0.25$. With this rule, the weight update is only applied when a burst occurs in the place cells. A burst is defined as any spike triplets that occur within 50 ms. In experiments, these post-synaptic bursts were associated with Ca^{2+} plateaus in place cells (Bittner et al., 2017) that lead to a large potentiation of synaptic strength with as few as 5 pairings. The connections of

M were initialized from a half-normal distribution $f(0, \sigma_{inp})$, where $\sigma_{inp} = 0.1$ and the signal amplitude was set to 1. M was bound between 0 and $5^*\sigma_{inp}$ during training.

3.2.3 *Audio processing for speech learning*

We used the numpy/python audio tools from (K. Kastner, 2019), adapted by (Sainburg, 2018), to process the audio WAVE file. We used the built-in functions to convert the audio file to a mel-scaled spectrogram and to invert it back to a waveform.

3.3 Results

3.3.1 *A reservoir network driven by oscillations*

We began with a basic implementation of our model where artificial oscillations served as input to a reservoir network (Fig. 16a) – in a later section, we will describe a more realistic version where recurrent networks generate these oscillations intrinsically. In this simplified model, two input nodes, but potentially more (Fig. 17), generate sinusoidal functions of different frequencies. These input nodes project onto a reservoir composed of a recurrent spiking neural network. This network is a conductance-based leaky integrate-and-fire (LIF) model (Destexhe, 1997) with balanced excitation and inhibition (C. van Vreeswijk & Sompolinsky, 1996). Every cell in the network is either strictly excitatory or inhibitory, thus respecting Dale's principle. The combination of N_{inp} input oscillators will generate a sequence of unique N_{inp}-dimensional vectors where the sequence lasts as long as the least common multiple of the inputs' individual periods (Vincent-Lamarre et al., 2016). For instance, two sine waves with

periods of 200 ms and 250 ms would create a multi-periodic input with a period lasting 1,000 ms. This effect can be viewed as a two-dimensional state-space where each axis is an individual sine wave (Fig.16b). Thus, multiplexed oscillations provide the network with inputs whose timescale largely exceeds that of individual units. When a reservoir network (N_{res} = 2,000) was injected with oscillations, excitatory and inhibitory populations modulated their activity over time, while the average input currents to individual neurons remained balanced (Fig.16c, top panel). To illustrate the benefits of oscillatory inputs on a reservoir, we designed a simple task where a network was trained to reproduce a target function consisting of a time-varying signal generated from low-pass filtered noise (Fig.16c, bottom panel). Simulations were split into a training and a testing phase. During the training phase, the network received a combination of two oscillatory inputs at 4 and 5 Hz. Synaptic weights from the reservoir to the read-out were adjusted using the regularized least-squares learning algorithm (Haykin, 2002; Sussillo & Abbott, 2009) adapted to spiking units (Nicola & Clopath, 2017) (see Methods). During the testing phase, synaptic weights were frozen and the network's performance was assessed by computing the Pearson correlation between the target function and the network's output. This correlation increased to 0.9 within the first 10 training epochs and remained stable thereafter (Fig.16d). By comparison, the output of a similar network with no oscillatory inputs remained uncorrelated to the target function. Thus, oscillatory inputs provided rich and reliable dynamics that enabled the reservoir to produce a target function that evolved over time in a precise manner. Next, we investigated the resilience of the network to structural perturbations where a number of individual neurons from the reservoir were "clamped" (i.e., held at resting potential) after train-

ing (Vincent-Lamarre et al., 2016). We trained a network for 10 epochs, then froze the weights and tested its performance on producing the target output. We then gradually clamped an increasing proportion of neurons from the reservoir. The network's performance decreased gradually as the percentage of clamped units increased (Fig.16e). Remarkably, the network produced an output that correlated strongly with the target function (correlation of 0.7) even when 10% of neurons were clamped. Further exploration of the model shows a wide range of parameters that yield high performances (Fig.17). Oscillatory inputs thus enabled reservoir networks to produce precise and repeatable patterns of activity under a wide range of modeling conditions. Next, we improved upon this simple model by developing a more biologically-inspired network that generated oscillations intrinsically.

3.3.2 *Endogenously generated oscillatory activity*

While our results thus far have shown the benefits of input oscillations when training a reservoir model, we did not consider their neural origins. To address this issue, we developed a model that replaces this artificial input with activity generated by an "oscillator" spiking network acting as a central pattern generator (Marder & Calabrese, 1996). To do so, we took advantage of computational results showing that sparsely connected networks can transition from an asynchronous to a periodic synchronous regime in response to a step current (Brunel, 2000; Brunel & Hansel, 2006; Thivierge, Comas, & Longtin, 2014), thus capturing *in vivo* activity (Buzsáki & Draguhn, 2004) (see Methods) (Fig.18a). The periodicity of the synchronous events could therefore potentially be used to biologically capture the effects of artificially generated sine waves. In simulations, we found that this transition was robust to both synaptic noise and neuronal clamping (Fig.19).

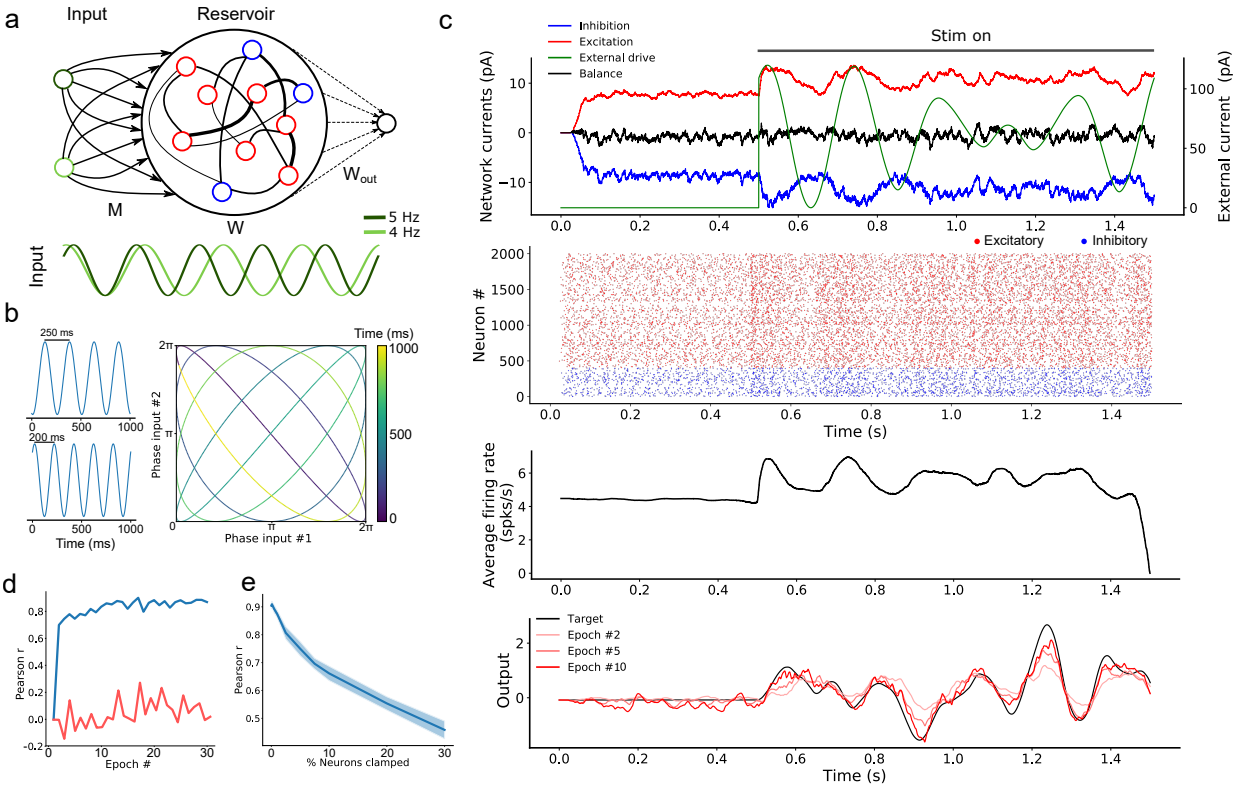

Figure 16: Oscillation driven spiking recurrent network as a reservoir computing model. **a** Schematic of the model's architecture. This implementation has two input units that inject sine-waves in a subset of the reservoir's neurons. M denotes the connections from the input units to the reservoir neurons. W denotes the recurrent connectivity matrix of the reservoir. W_{out} denotes the trainable connections from the excitatory reservoir neurons to the readout unit. **b** Example showing how two sine waves of different frequencies can be combined to generate a two-dimensional function with a period longer than either sine wave. **c** Sample average current per neuron, raster and average instantaneous firing rate respectively. The external drive is only delivered to a subset of reservoir units ($p_{\mathrm{inp}} = 0.3$) for each input unit. Bottom: Output of the readout unit during testing trials (learning rule turned off) interleaved during training. **d** Pearson correlation between the output of the network and the target function as a function of the number of training epochs when driven with (blue) and without (red) multi-periodic input. **e** Pearson correlation between the output and the target function as neurons of the reservoir are clamped.

Further, the frequency of synchronized events could be modulated by adjusting the strength of the step current injected in the network, with stronger external inputs leading to a higher frequency of events

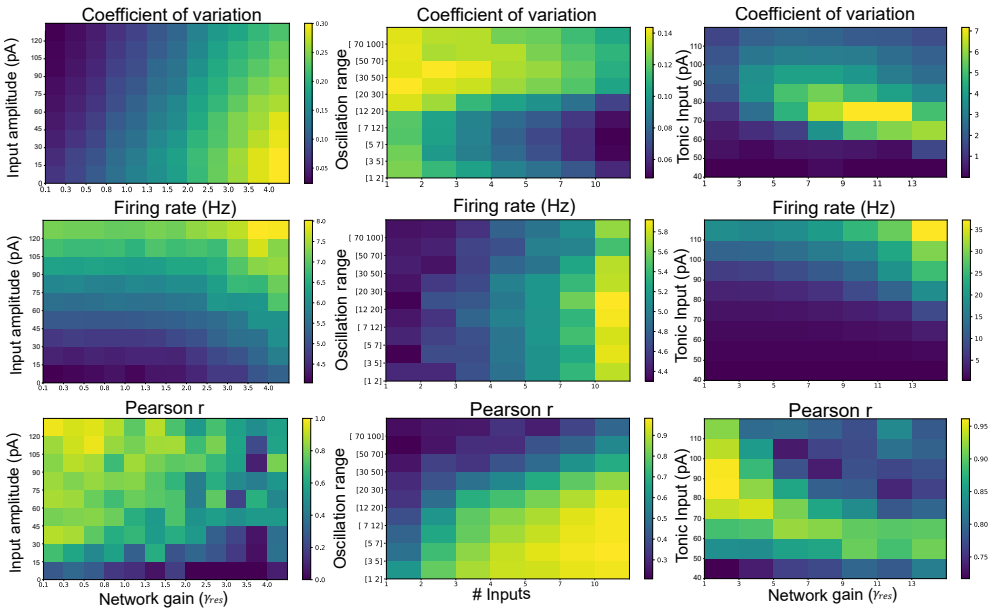

Figure 17: Parameter exploration for training a reservoir network. The top row shows the average inter-trial coefficient of variation (CV) with time bins of 1 ms, the middle row shows the average firing rate, and the bottom row shows the average correlation between the output of the network and the target output. *Input amplitude* is the strength of the input projecting to the reservoir's units. *Network gain* is the strength of the recurrent connections between the reservoir's units. *Oscillation range* is the range from which each input unit's sine wave period is randomly drawn. *# Inputs* is the number of different input units projecting to the reservoir. *Tonic input* is the strength of the constant current injected to each of the reservoir's units.

(Fig.19). Thus, oscillator networks provide a natural neural substrate for input oscillations into a recurrent network.

From there, we formed a model where three oscillator networks fed their activity to a reservoir network (Fig.18b). These oscillator networks had the same internal parameters except for the inhibitory decay time constants of their recurrent synapses (τ_{in} = 70, 100 and 130 ms for each network) thus yielding different oscillatory frequencies (Figs. 20,21). In order to transition from an asynchronous to a synchronous state, the excitatory neurons of the oscillator networks received a step current.

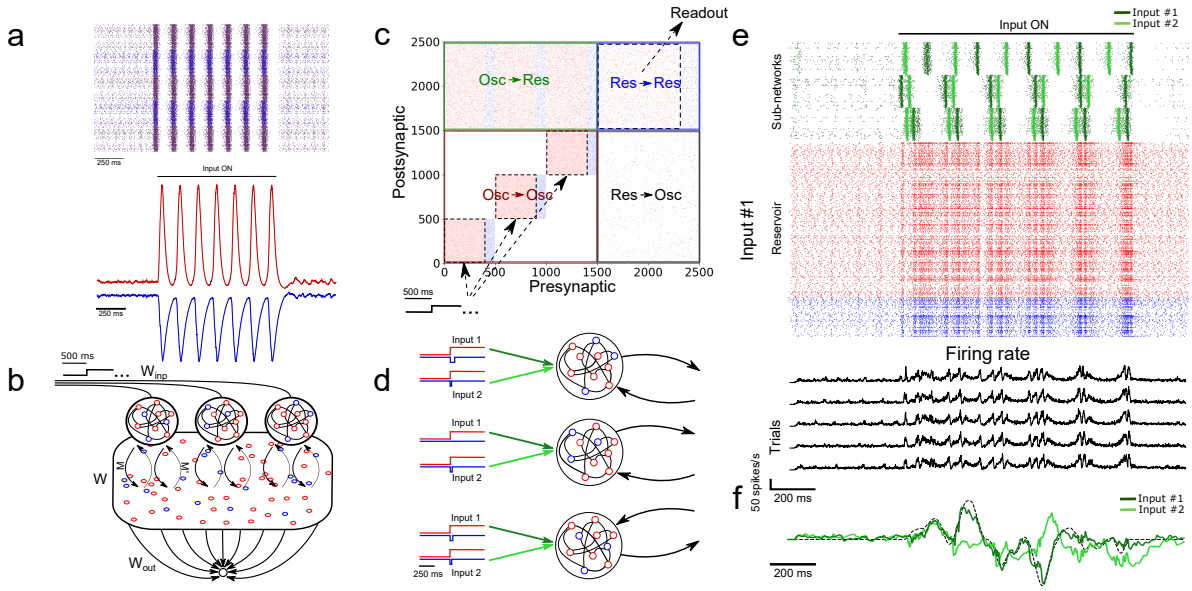

Figure 18: Reservoir driven with intrinsically generated oscillations. **a** Top: Sample rasters from one network on five different trials. Bottom: Average and standard error of the excitatory (red) and inhibitory (blue) conductance of a stable network on 5 different trials. The network is asynchronous when the external drive is off, and becomes periodic when turned on. **b** Architecture of the augmented model. As in 16a, W and W_{out} denote the recurrent and readout connections, respectively. M is the connection matrix from the input to the reservoir, except that the input units are now replaced by networks of neurons. M' denotes the feedback connections from the reservoir to the oscillatory network. W_{inp} denotes the connections providing the tonic depolarization to the oscillatory networks. **c** Connectivity matrix of the model. The external drive is provided solely to the oscillatory networks that project to the reservoir network that in turn projects to the readout unit. **d** External inputs provided to the oscillatory networks with varied inhibitory transients. The inhibitory transients to the three oscillatory networks were 100, 0 and 50 ms (Input 1), and 10, 30 and 20 ms (Input 2), respectively. **e** Top: Sample activity of the oscillatory networks (green) on two separate trials with different inputs, and the reservoir for one trial (input #1). Bottom: PSTH of the network's activity on five different trials with input #1. **f** Post-training output of the network, where input #1 was paired with the target, but input #2 was not.

The full connectivity matrix of this large model is depicted in Fig.18c. As shown, the oscillator networks send sparse projections (with a probability of 0.3 between pairs of units) to the reservoir units. Only the excitatory neurons of the reservoir network project to the readout units. Only weak feedback from the reservoir to the oscillators was

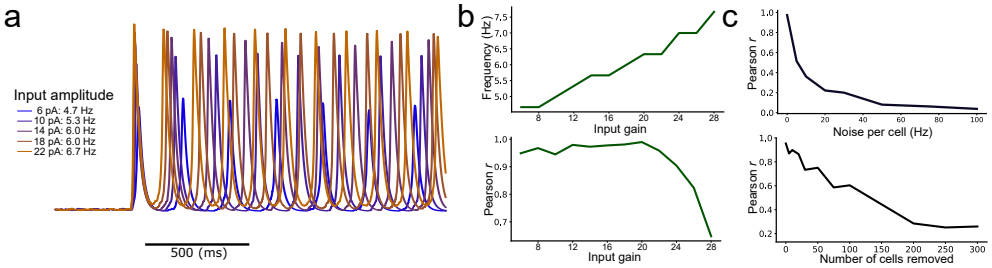

Figure 19: Autonomous production of repeatable periodic activity. **a** Network activity as a function of the strength of the external drive. The frequency scales with the strength of the tonic input. **b** Top: Average frequency of the network's activity as a function of the external drive. Bottom: Pearson correlation between multiple trials as a function of the amplitude of the external drive. **c** Pearson correlation between multiple trials for a given network as a function of the rate of noisy synaptic input per neuron (top) and as a function of the proportion of clamped neurons (bottom).

included - in supplementary simulations, we found that strong feedback projections desynchronized the oscillator networks (Figs. 19,22). A sample of the full model's activity is shown in Fig.18e. Both the oscillator and the reservoir networks showed asynchronous activity until a step current was injected into the excitatory units of the oscillator networks. In response to this step current, both oscillator and reservoir networks transitioned to a synchronous regime. The model reverted back to an asynchronous regime once the step current was turned off. To illustrate the behaviour of this model, we devised a "cued" task similar to the one described above, where the goal was to reproduce a random time-varying signal. When learning this signal, however, the oscillator networks received a cue ("Input 1") consisting of a combination of excitatory step current and transient inhibitory input (Fig.18d) that alters the relative phase of the input oscillators, but not their frequency. Following 10 epochs of training, we switched to a testing phase and showed that the model closely matched the target signal (Fig.18f). Crucially, this behaviour of the model was specific to the cue provided during training: when a different, novel cue (shaped by inhibitory transients) was presented to the network

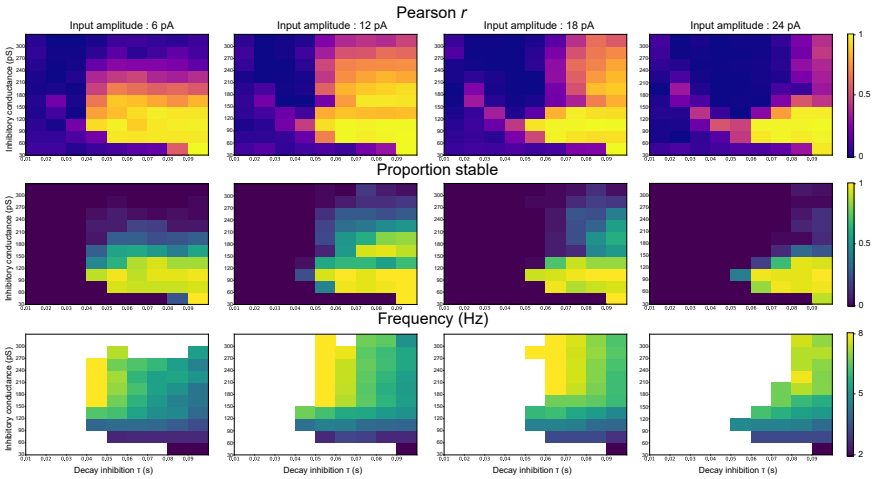

Figure 20: Parameter exploration of the network's synchronous activity (inhibition conductance and kinetics). Each column represents a different strength of the external input step, and each heatmap shows different combinations of the conductance and synaptic time-constant of inhibitory inputs. The top row shows the average Pearson correlation between the output and the target function for 5 trials for each of the 50 random network initializations tested per condition. The middle row shows which proportion of the networks are considered as stable. The bottom row shows the average frequency of the periodic activity of the stable networks.

("Input 2"), an different output was produced (Fig.18f). In sum, the model was able to learn a complex time-varying signal by harnessing internally-generated oscillations that controlled the ongoing activity of a reservoir network. In the following section, we aimed to further explore the computational capacity of the model by training a reservoir on multiple tasks in parallel.

3.3.3 *An artificial network that learns to multitask*

To explore the ability of the model to learn two tasks concurrently, we reverted to our initial model with artificial oscillations, allowing for a

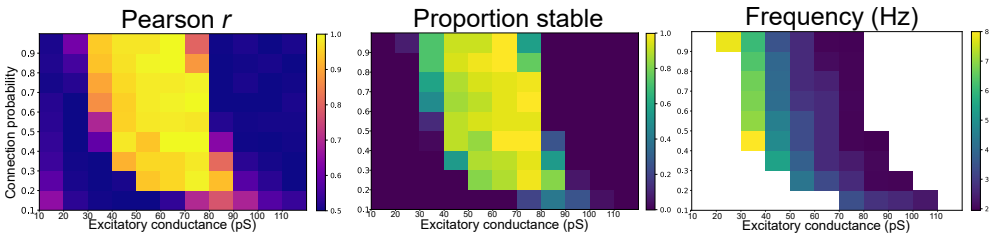

Figure 21: Parameter exploration of synchronous activity (excitatory conductance and p). The sparseness of the reservoir and the conductance of the excitatory connections were systematically varied to observe their impact on the network's regime. *Left*: Pearson correlation between the output and the target function for 5 trials for each of the 50 random network initializations tested per condition. *Middle*: proportion of networks that are considered stable. *Right*: average frequency of the periodic activity of the stable networks.

more principled control of the input injected to the reservoir network. The oscillatory input consisted of three sine waves of different frequencies (Fig.23a) (more inputs lead to richer dynamics, Fig.17) . We trained this model on two different motor control tasks that required the network to combine the output of two readout units in order to draw either a circle or a star in two dimensions. Here, each of the outputs corresponded to x- and y-coordinates, respectively. The phase of the oscillations ("Input 1" vs. "Input 2") were individually paired with only one of the two tasks in alternation (Fig.23a). We employed a principal component analysis (PCA) to visualize the activity of the reservoir before and during training (Fig. 23b). Before injecting the oscillatory inputs, the network generated spontaneous activity that occupied a limited portion of the state space (Fig. 23b, circle). During training, the oscillatory inputs were turned on, resulting in different trajectories depending on the relative phase of the oscillations. The network thus displayed a distinct pattern of activity for each of the two tasks. Viewed in two dimensions, the outputs of the network rapidly converged to a circle and a star that corresponded to each of the two target shapes when given each respective input separately

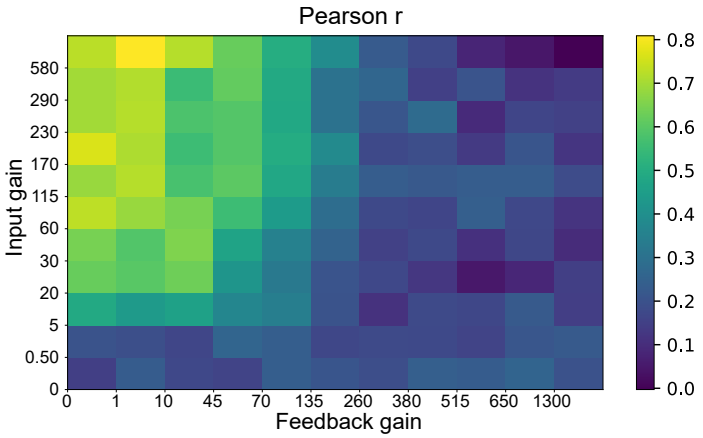

Figure 22: Impact of reservoir feedback to the oscillatory networks. The models can perform well when the gain of the feedback connections from the reservoir to the oscillatory networks are scaled up to a limit. The effect of the feedback gain appears to be independent of the strength of the tonic input of the step function to the oscillatory networks.

(Fig.23c). These shapes were specific to the particular phase of the oscillatory input in a condition where we presented a randomly-chosen phase configuration to the network, the output did not match either of the trained patterns (Fig.23c). Finally, we tested the ability of the model to learn a number of target signals varying in duration and frequency. We generated a number of target functions consisting of filtered noise (as described previously), and varied their duration as well as the cut-off frequencies of band-pass filtering. The network performed optimally for tasks with relatively low frequency (<30 Hz) and shorter duration (<5 seconds), and had a decent performance for even longer targets (e.g, correlation of r=0.5 for a time of 7 seconds and a frequency below 30 Hz). (Fig. 23d). In sum, the model was able to learn multiple tasks in parallel based on the phase configuration of the oscillatory inputs to the reservoir units. The range of target signals that could be learned was dependent upon their duration and frequency. The next section will investigate another aptitude of the network, where a target signal can be rescaled in time without further training.

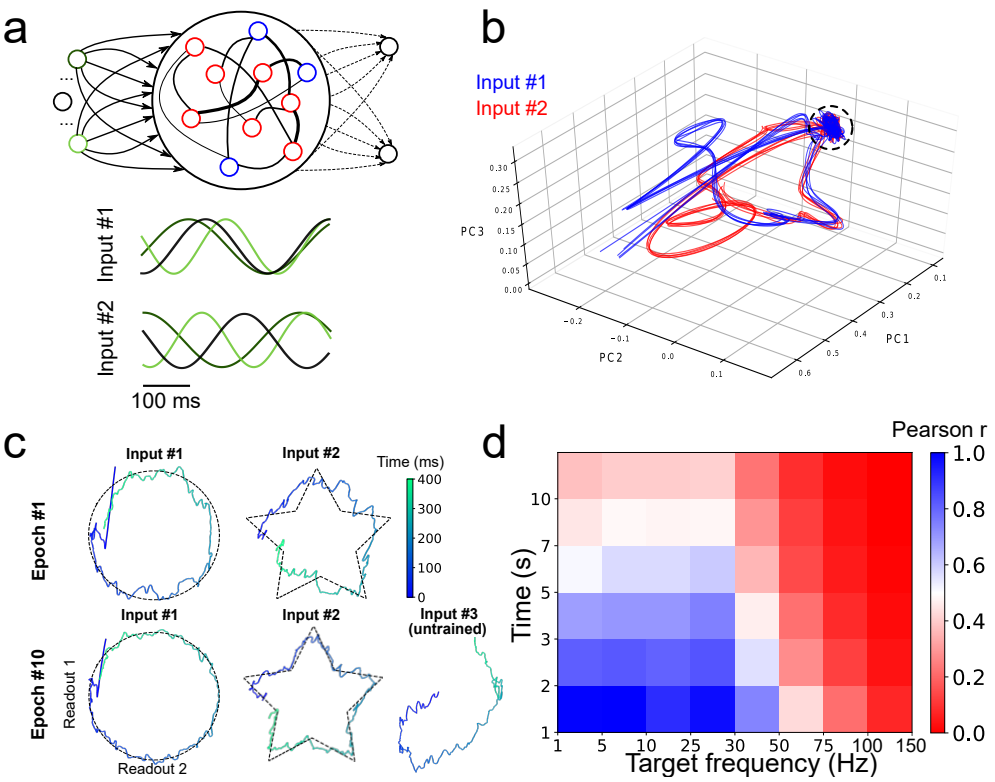

Figure 23: Parallel training of multiple tasks. **a** Schematic of the model's architecture with an additional readout unit. An example of possible phase shifted inputs is shown at the bottom. **b** Trajectory of the reservoir on the first three components of a PCA. Without an external input, the network is spontaneously active and constrained in a subspace of the state-space (depicted by the circle). Upon injection of the input, the reservoir's activity is kicked into the trajectory related to the input. **c** Output of the two readout units after the first and tenth training epochs. **d** Heat map showing the performance of the model on tasks with different lengths and frequencies.

3.3.4 *Temporal rescaling of neuronal activity*

A key aspect of many behavioural tasks based on temporal sequences is that once learned they can be performed faster or slower without additional training. For example, when a new word is learned, it can be spoken faster or slower without having to learn the different speeds separately. We propose a straightforward mechanism to rescale a learned temporal sequence in the model. Because the activity of the model strongly depends upon the structure of its oscillatory inputs, we conjectured that the model may generate a slower or

faster output by multiplying the period of the oscillatory inputs by a common factor. Biologically, such a factor might arise from afferent neural structures that modulate oscillatory activity (Brunel, 2000). Due to the highly non-linear properties of the network, it is not trivial that rescaling the inputs would expand or compress its activity in a way that preserves key features of the output (Goudar & Buonomano, 2018). To test the above mechanism, we trained a reservoir network receiving sine wave inputs to produce a temporal sequence of low-pass filtered random activity. After the pair of input-target was trained for 10 epochs, we tested the network by injecting it with sine waves that were either compressed or expanded by a fixed factor relative to the original inputs (Fig.24a). To evaluate the network's ability to faithfully replay the learned sequence, we computed the Pearson correlation between the output of the network (Fig.24b) and a compressed or expanded version of the target signal. Performance degraded gradually with inputs that were expanded or contracted in time relative to the target signal (i.e, as the rescaling factor moved further away from 1) (Fig.24c). Further, performance degraded more slowly beyond a rescaling factor of 1.5, particularly when input noise was absent, suggesting some capacity of the network to expand the target signal in time (Fig. 26). This result offered a qualitative match to experimental findings (Hardy et al., 2018) and the performance of the network was robust to the addition of random jitter in the phase of the input oscillations (Fig.25).

In sum, rescaling the speed of the input oscillations by a common factor lead to a corresponding rescaling of the learned task, with compressed neural activity resulting in more error than expanded activity. The next section examines some of the underlying features of activity in a reservoir network driven by oscillatory inputs.

3.3.5 *Temporal selectivity of artificial neurons*

A hallmark of temporal processing in brain circuits is that some subpopulations of neurons increase their firing rate at specific times during the execution of a timed task (Bakhurin et al., 2017; Harvey et al., 2012; Mello et al., 2015). To see whether this feature was present in the model, we injected similar oscillatory inputs as above (10 input units between 5 and 10 Hz for 1 second) for 30 trials. To match experimental analyses, we convoluted the firing rate of each neuron from the reservoir with a Gaussian kernel (s.d. = 20 ms), averaged their activity over all trials, and converted the resulting values to a z-score. To facilitate visualization, we then sorted these z-scores by the timing of their peak activity. We retained only the neurons that were active during the simulations (71%). Results showed a clear temporal selectivity whereby individual neurons increased their firing rate at a preferred time relative to the onset of each trial (Fig.24d). These "selectivity peaks" in neural activity were maintained in the same order when we expanded or contracted the input oscillations by a fixed factor (Fig.24a), and sorted neurons based on the original input oscillations (Fig.24d), thus capturing recent experimental results (Mello et al., 2015). To shed light on the ability of simulated neurons to exhibit temporal selectivity, we examined the timing of excitatory and inhibitory currents averaged across neurons of the reservoir. We then aligned these currents to the timing of selectivity peaks and found elevated activity around the time of trial-averaged peaks (Fig.24e). Therefore, both the input E/I currents and the external inputs drive the activity of the neurons near their peak response, showing that an interaction between intrinsic and external sources drives the temporal selectivity of individual neurons. In sum, neurons from the reservoir

show sequential patterns of activity by leveraging a combination of external drive and recurrent connections within the network. Next, we examined the ability of the model to learn a naturalistic task of speech production.

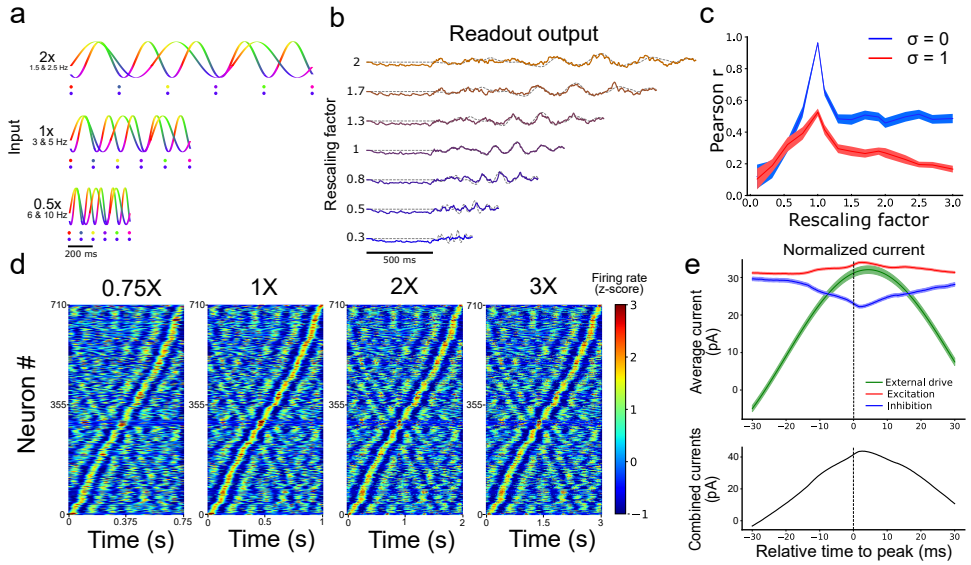

Figure 24: Temporal rescaling of the network's activity. **a** Activity of two input units with different rescaling factors. Each phase is represented in a different color. The two dots shown for the different velocities are spaced at a constant proportion of the whole duration, and show that the phase alignment of the two oscillators is preserved across different rescaling factors. **b** Output of the network for different rescaling factors. The target is represented by the grey dashed line. **c** Pearson correlation between the output and the target function for different rescaling factors and different input noise variance (σ). **d** Spiking activity of the reservoir at 0.75X, 1X, 2X, and 3X sorted based on the peak of the activity at the original scaling (1X). **e** Average excitatory (red), inhibitory (blue), and external input (green) received by each cell aligned on their peak firing rate (at t = 0 ms). The combined inputs (black, external + excitation - inhibition) is represented at the bottom.

3.3.6 *Learning Natural Speech, Fast and Slow*

In a series of simulations, we turned to a biologically and behaviourally relevant task of natural speech learning. This task is of particular relevance to temporal sequence learning given the precise yet flexible nature of speech production: once we learn to pronounce a word,

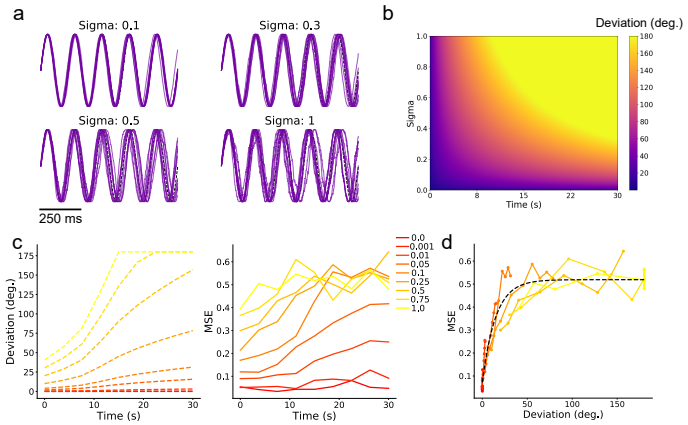

Figure 25: Network performance with desynchronized inputs. **a** Four simulations of 10 trials each with different σ_ϕ with a random walk process added to the phase of a sine wave. **b** Analytical standard deviation (in degrees) around the expected phase of a deterministic signal as a function of time at different values of σ_ϕ. **c** Left: Analytical estimation of the standard deviation in degrees with the different σ_ϕ used for the network simulation. Right: corresponding MSE between the output and target of the readout unit for the different conditions tested. **d** Relationship between the deviation in degree and the MSE. An exponential function was fit to the data (R^2=0.92).

it is straightforward to alter the speed at which this word is spoken without the need for further training. We thus designed a task where an artificial neural network must learn to utter spoken words in the English language and pronounce them slower or faster given the appropriate input, without retraining. To train a network on this task, we began by extracting the waveform from an audio recording of the word "reservoir" and converting this waveform to a spectrogram (Fig.27a). We then employed a compression algorithm to bin the full range of frequencies into 64 channels spanning a range from 300 Hz to 8 kHz (see Methods). Each of these channels were mapped onto an individual readout unit of the model. Synaptic weights of the reservoir to the readout were trained to reproduce the amplitude of the 64 channels over time. The output spectrogram obtained from the readout units was converted to an audio waveform and compared to the target waveform. Following training, the network was

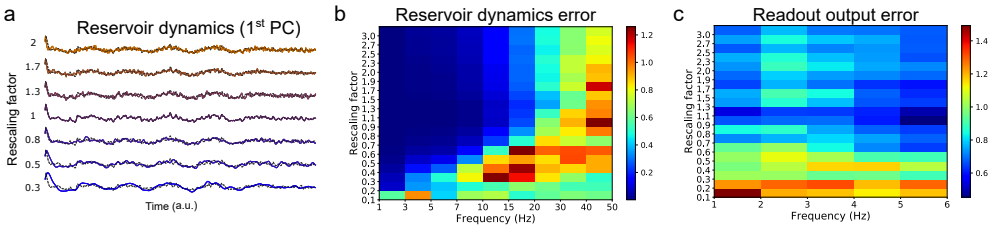

Figure 26: Network dynamics and performance with temporal rescaling. **a** Activity of the reservoir projected on the first principal component for different rescaling factors. The black dashed line represents the activity without rescaling. The traces displayed have been scaled back to the original velocity. **b** Normalised error of the first principal component of the reservoir's activity for different rescaling factors and frequency bands. **c** Normalised error of the network's output (MSE between the output and the target divided by the total variance of the target function) for different rescaling factors and frequency bands.

able to produce a waveform that closely matched the target word (Fig.27a). To examine the ability of the network to utter the same word faster or slower, we employed the rescaling approach described earlier, where we multiplied the input oscillations by a constant factor (Fig.24a). Our model was able to produce both faster and slower speech than what it had learned (Fig.27b). Scaling the outputs back to the original speed showed that the features of the spectrogram were well replicated (Fig.27c). The correlation between the rescaled outputs and target signal decreased as a function of the rescaling factor (Fig.27d), in a manner similar to the above results on synthetic signals (Fig.24c). Results thus suggest that mixing oscillatory inputs enabled a reservoir network to acquire and rescale temporal sequences obtained from natural speech. In the final section below, we employed our model to capture hippocampal activity during a well-studied task of spatial navigation.

3.3.7 *Temporal sequence learning during spatial navigation*

Thus far, we have modelled tasks where units downstream of the reservoir are learning continuous signals in time. In this section, we

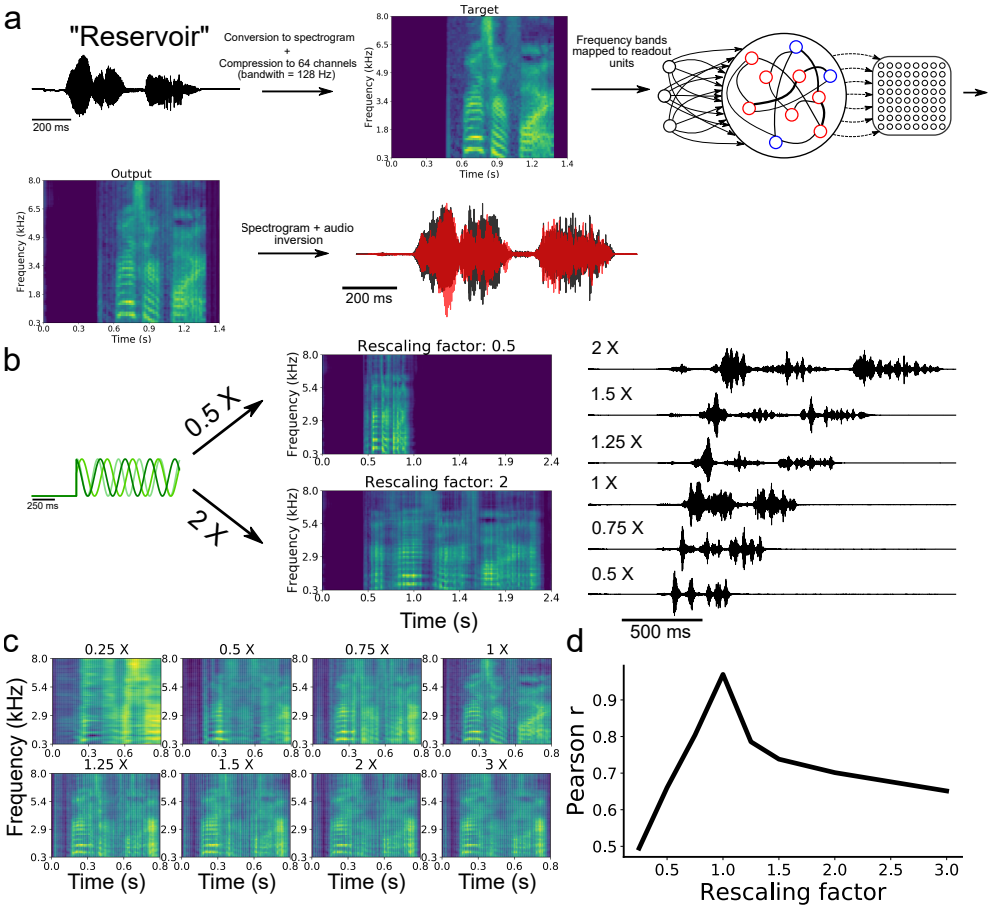

Figure 27: Speech learning and production with temporal rescaling. **a** Workflow of the transformation and learning of the target audio sequence. **b** The input frequencies are either sped-up or slowed-down in order to induce temporal rescaling of the speed of execution of the task. **c** The rescaled output spectrograms are scaled back to the original speed in order to compare them to the target spectrogram. **d** Average correlation between the output and target of all channels of the spectrograms for the different rescaling factors.

turned to a task of spatial navigation that required the model to learn a discrete sequence of neural activity. A wealth of experiments shows that subpopulations of neurons become selectively active for specific task-related time intervals. A prime example is seen in hippocampal theta sequences (Foster & Wilson, 2007) that emerge during spatial navigation in rodents, where individual place cells (O'Keefe, 1976) increase their firing rate at a given location in space (place fields). During spatial navigation, the hippocampus shows oscillatory activity in

the theta range (4-12 Hz), likely originating from both the medial septum (Dragoi, Carpi, Recce, Csicsvari, & Buzsáki, 1999) and within hippocampus (Goutagny, Jackson, & Williams, 2009; Traub, Miles, & Wong, 1989) . In a series of simulations, we examined how neurons in a reservoir network may benefit from theta oscillations to bind and replay such discrete place cells sequences. We designed a reservoir (associated with area CA1, Foster & Wilson, 2007) that received inputs from multiple input oscillators (CA3, (Montgomery, Betancur, & Buzsáki, 2009)) (Fig.28a,b). We randomly selected 10 excitatory units within the reservoir and labelled them as "place cells", and removed synapses between excitatory cells due to the lack of reccurrent connections between excitatory neurons in CA1 (Deuchars & Thomson, 1996). To simulate the response of place cells to an environmental input indicating the spatial location of the animal (Frank, Brown, & Wilson, 2000), we depolarized these cells by an oscillating input at 10 Hz for 600 ms with a specific onset that differed across neurons in order to capture their respective place fields (assuming a fixed spatiotemporal relation of 100 ms = 5 cm on a linear track). In this way, the sequential activation of place cells from the reservoir network mimicked the response of CA1 neurons to an animal walking along a linear track (Fig.28c). To capture the effect of theta oscillations on CA1 activity, all neurons from the reservoir were driven by a combination of multiple oscillating inputs where the frequency of each input was drawn from a uniform distribution in the range of 7-9 Hz. Connections from the input units and the place cells within the reservoir were modified by a synaptic plasticity rule (see Methods). As expected from the input oscillators, mean population activity of the reservoir network exhibited prominent theta activity (Fig.28d). To assess the baseline performance of the model, we ran an initial simulation with oscillatory

inputs but no synaptic plasticity or place fields (i.e., no environmental inputs to the place cells). All place cells of the model remained silent (Fig.28g). Next, we ran a training phase simulating a single lap of the virtual track lasting 5 seconds, where place cells received oscillatory inputs (CA3) as well as a depolarizing oscillation (10 Hz) whenever the cell entered its place field. During this lap, individual place cells entered their respective field only once. We assessed the performance of the model during a testing phase where both synaptic plasticity and depolarizing oscillations were turned off. During the testing phase, place cells yielded a clear sequence of activation that matched the firing pattern generated during training (Fig.28g). Thus, place cell activity was linked to the phase of the oscillatory inputs after a single lap of exploration. Going further, we explored two key aspects of place cell activity in the reservoir that are reported in hippocampus, namely phase precession and rapid replay. During phase precession, the phase of firing of place cells exhibits a lag that increases with every consecutive cycle of the theta oscillation (Foster & Wilson, 2007; O'Keefe & Recce, 1993). We examined this effect in the model by extracting the instantaneous phase of firing relative to the global firing rate filtered between 4-12 Hz. The activity of individual place cells from the reservoir relative to theta activity exhibited an increasing phase lag characteristic of phase precession (Fig.28e,f). A second feature of hippocampal activity is the rapid replay of place cells during rest and sleep in a sequence that mirrors their order of activation during navigation (Lee & Wilson, 2002). This replay can arise in either a forward or reverse order from the original sequence of activation (Foster & Wilson, 2006). We compressed (factor of 0.15) the CA3 theta oscillations injected in the reservoir during training, resulting in rapid (50-55 Hz) bursts of activity (Fig.28h). These fast oscilla-

tions mimicked the sharp-wave ripples that accompany hippocampal replay (Lee & Wilson, 2002). In response to these ripples, place cells of the reservoir exhibited a pattern of response that conserved the order of activation observed during training (Fig.28h). Further, a reverse replay was obtained by inverting the ripples (that is, reversing the order of the compressed sequence) presented to the reservoir (Fig.28h). In sum, oscillatory inputs allowed individual neurons of the model to respond selectively to external inputs in a way that captured the sequential activation and replay of hippocampal place cells during a task of spatial navigation.

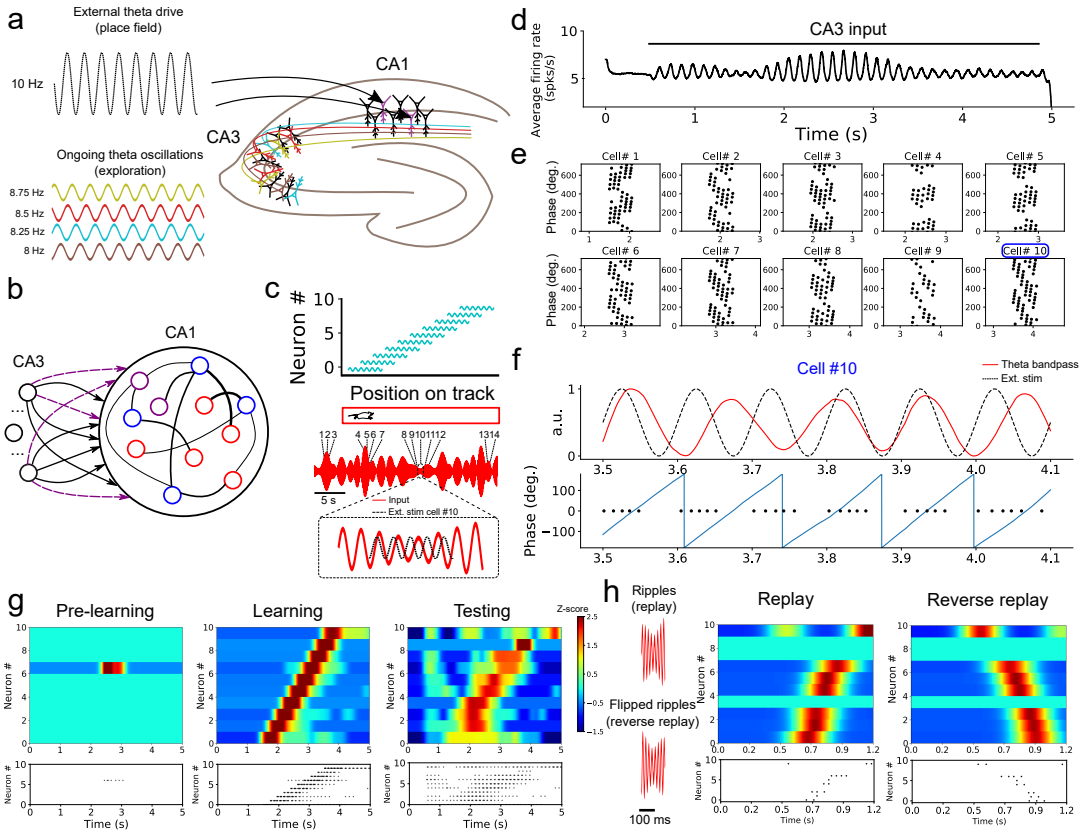

Figure 28: Formation of place cell sequences and replay. **a** Subpopulations of CA3 cells oscillating in the theta range and projecting to CA1. CA1 place cells are driven by a slightly faster oscillating input upon entering their place field. **b** RC implementation of the phenomenological model. The input layer is composed of oscillatory units (CA3) and CA1 is modelled by a reservoir where 10 excitatory units (purple) are randomly selected as place cells. The connections from the input to the place cells are subject to training. **c** Top: Each of the 10 reservoir place cells were driven by a depolarizing oscillating input in a temporal sequence analogous to a mouse moving along a linear track. Bottom: Resulting theta frequency of the combined oscillations (red). Each number shows a place cell activated at a given time along the ongoing theta input. The multi-periodic input from CA3 guarantees that each place cell is activated with a unique combination of the input, following the sequence in which the cell is active. **d** Mixing CA3 inputs generates a visible theta oscillation in the CA1 reservoir. **e** Spike times of the ten place cells in relation to the phase of the population theta activity. Each dot represents a spike at a certain time/position. Each cell shows a shift towards earlier theta phase as the animal moves along its place fields. **f** Activity of cell #3 upon entering its place field. Top: theta band-passed activity of the reservoir (red) and place field related input (dashed black). Bottom: phase of the theta oscillation (grey) and spike times. **g** Top shows a heatmap of the place cell activity and bottom shows the spike raster during each phase of training. Before training: All place cells are silent. During training: place cells are depolarized upon entering their place field. After training: a similar sequence is evoked without the external stimulation used during training. **h** Rescaling the input (factor of X0.15) leads to a high-frequency input reminiscent of ripples. A compressed version of the sequence learned in **g** is evoked, and a reversed sequence is evoked when a reverse "ripple" is injected in the reservoir.

3.4 Discussion

3.4.1 *Summary of results*

Taken together, our results suggest that large recurrent networks can benefit from autonomously generated oscillatory inputs in order to learn a wide variety of artificial and naturalistic signals, and exhibit features of neural activity that closely resemble neurophysiological experiments. One series of simulations trained the model to replicate simple shapes in 2D coordinates. Based upon the structure of its oscillatory inputs, the model flexibly switched between two shapes, thus showing a simple yet clear example of multitasking with a recurrent network. When we modulated the period of input oscillations delivered to neurons of the reservoir, the model was able to produce an output that was faithful to the target signal, but sped up or slowed down by a constant factor (Hardy et al., 2018; Mello et al., 2015). Oscillations served to train a recurrent network that reproduced natural speech and generated both slower and faster utterances of natural words with no additional training. Using further refinements of the model, we employed this principle of oscillation-driven network to capture the fast replay of place cells during a task of spatial navigation. Below we discuss the biological implications of our model as well as its applications and limitations.

3.4.2 *Biological relevance and predictions of the model*

Despite some fundamental limitations common to most computational models of brain activity, our approach was designed with several key features of living neuronal networks, including spiking neurons, Dale's principle, balanced excitation/inhibition, a heterogeneity of neuronal and synaptic parameters, propagation delays, and conductance-

based synapses (Ingrosso & Abbott, 2019; Sussillo & Abbott, 2009; C. van Vreeswijk & Sompolinsky, 1996). Further, and most central to this work, our model included neural oscillations along a range of frequencies that closely matched those reported in electrophysiological studies (Buzsáki & Draguhn, 2004; Yuste et al., 2005). Although there is an abundance of potential roles for neural oscillations in neuronal processing, much of their function remain unknown (Wang, 2010). Here, we proposed that multiple heterogeneous oscillations may be combined to generate an input whose duration greatly exceeds the time-course of any individual oscillation. In turn, this multiplexed input allows a large recurrent network operating in the chaotic regime to generate repeatable and stable patterns of activity that can be read out by downstream units. It is well established that central pattern generators in lower brain regions such as the brain stem and the spinal cord are heavily involved in the generation of rhythmic movements that match the period of simpler motor actions (e.g. walking or swimming) (Marder & Bucher, 2001). From an evolutionary perspective, it is compelling that higher brain centers would recycle the same mechanisms (Yuste et al., 2005) to generate more complex and non-repetitive actions (Churchland et al., 2012; Rokni & Sompolinsky, 2011). In this vein, (Churchland et al., 2012) showed that both periodic and quasi-periodic activity underlie a non-periodic motor task of reaching. Our model provides a framework to explain how such activity can be exploited by living neuronal networks to produce rich dynamics whose goal is to execute autonomous aperiodic tasks. While our model shows that oscillatory networks can generate input oscillations that control the activity of a reservoir (Fig.18), the biological identity of these oscillatory networks is largely circuit-dependent and may originate from either intrinsic or extrinsic sources.

In the hippocampus, computational (Traub et al., 1989) and experimental (Goutagny et al., 2009) findings suggest an intrinsic source to theta oscillations. Specifically, studies raise the possible role of CA3 in forming a multi-periodic drive consisting of several interdependent theta generators that activate during spatial navigation (Montgomery et al., 2009). Similarly, the neural origin of the tonic inputs controlling the activity of the oscillatory networks is not explicitly accounted for in our simulations. However, it is well established that populations of neurons can exhibit bistable activity with UP-states lasting for several seconds (Wang, 2016) that could provide the necessary input to drive the transition from asynchronous to synchronous activity in oscillatory networks. Going beyond an *in silico* replication of neurophysiological findings, our model makes two empirically testable predictions. If one was to experimentally isolate the activity of the input oscillators, one could show that: (*i*) a key neural signature of a recurrent circuit driven by multi-periodic oscillations is the presence of inter-trial correlations between the phase of these oscillations; and (*ii*) the period of the input oscillators should appear faster or slower to match the rescaling factor of the network. This correspondence between the input oscillations and temporal rescaling is a generic mechanism behind the model's ability to perform a wide variety of tasks, from spatial navigation to speech production.

3.4.3 *Related models*

Our approach was inspired by predecessors in computational neuroscience. Mixing multiple oscillations as a way to generate long sequences of non-repeating inputs was first introduced by (Miall, 1989), with a model of interval timing relying on the coincident activation of multiple oscillators of different beats. This idea served as a basis

for the striatal beat frequency model (Matell & Meck, 2004), where multiple cortical regions are hypothesized to project to the striatum which acts as a coincidence detector that encodes timing intervals. A similar mechanism was also suggested for the representation of space by grid cells in rodents (Fiete et al., 2008). Grid cells have periodic activation curves spanning different spatial periods (Moser, Kropff, & Moser, 2008), and their activation may generate a combinatorial code employed by downstream regions to precisely encode the location of the animal in space (Fiete et al., 2008). Other studies have suggested that phase precession during spatial navigation could originate from a dual oscillator process (Burgess, Barry, & O'Keefe, 2007; O'Keefe & Recce, 1993). Along this line, a recent model of the hippocampus uses the interference between two oscillators to model the neural dynamics related to spatial navigation (Nicola & Clopath, 2019). Although this model shares similarities with ours, a fundamental difference is that our model uses the phase of combined oscillators to create a unique input at every time-step of a task, whereas their model relies on the beat of the combined frequencies. Additionally, in our model, increasing the frequency of input oscillations by a common factor leads to compressed sequences of activity. By comparison, in the model of (Nicola & Clopath, 2019), sequences are compressed by removing an extrinsic input oscillator. More experimental data will be needed to support either model.

3.4.4 *Limitations and future directions*

In our model, periodic activity was readily observable in the reservoir dynamics due to the input drive (e.g. Fig.16c or Fig.28d). However, the architecture of our model represents a simplification of biological networks where several intermediate stages of information process-

ing occur between sensory input and behavioural output. Oscillatory activity resulting from a multi-periodic drive might occur in one, but not necessarily all stages of processing. Further work could examine this issue by developing a multi-layered network of reservoirs; such a hierarchical organization may have important computational benefits (Gallicchio, Micheli, & Pedrelli, 2017). In the spatial navigation task, we ensured that the location of the animal was perfectly correlated with the time spent in the place field of each cell. This is, of course, an idealized scenario that does not account for free exploration and variable speed of navigation along a track. These factors would decorrelate the spatial location of the animal and the time elapsed in the place field. Hence, further work would benefit from a more ecologically-relevant version of the navigation task. This new version of the task might aim to capture how the time spent in a given place field impacts the link between the activity of place cells and theta oscillations (R. Schmidt et al., 2009) . Finally, our task of speech production was restricted to learning a spectrogram of the target signal. This simplified task did not account for the neural control of articulatory speech kinetics, likely involving the ventral sensory-motor cortex (Anumanchipalli, Chartier, & Chang, 2019; Conant, Bouchard, Leonard, & Chang, 2018).

Applications

Our modeling framework is poised to address a broad spectrum of applications in machine learning of natural and artificial signals. With recent advances in reservoir computing (Salehinejad et al., 2017) and its physical implementations (Tanaka et al., 2018), our approach offers an alternative to using external arbitrary time-varying signals to control the dynamics of a recurrent network. Our model may also

be extended to neuromorphic hardware, where it may benefit chaotic networks employed in robotic motor control (Folgheraiter et al., 2019). Finally, our model is, to our knowledge, the first to produce temporal rescaling of natural speech, with implications extending to conversational agents, brain-computer interfaces, and speech synthesis. Overall, our model offers a compelling theory for the role of neural oscillations in temporal processing. Support from additional experimental evidence could impact our understanding of how brain circuits generate long sequences of activity that shape both cognitive processing and behaviour.

4 CHAPTER 4: GENERAL DISCUSSION

RC models have been gaining attention as plausible models of cortical circuits across a wide range of tasks. We were mainly interested in models operating in a chaotic regime which has been hypothesized to occur in recurrent cortical circuits (Sompolinsky et al., 1988; C. van Vreeswijk & Sompolinsky, 1996). We show in the first chapter that an aspect of such networks had been largely ignored: how do they tolerate structural perturbations? We showed that some mechanisms that are able to tame chaos in such networks do not have the same properties toward small alterations to the network parameters, such as removing one neuron out of thousands. This finding has implications about the type of autonomous activity that chaotic RNNs can produce, and raises doubt about the possibility that the brain works in ways compatible with the RC framework. However, our work suggests a workaround this limitation, with oscillation driven networks. Whereas it is still unknown how the brain produces complex aperiodic patterns of neural activity, simple and periodic dynamics are fairly well understood (Hooper, 2001; Yuste et al., 2005). We showed that by mixing oscillations with different periods in a RNN, some rich and repeatable neural trajectories can be evoked and used by downstream neurons (or networks of neurons) to learn a wide variety of tasks. We showed that with realistic reservoirs implementing a wide array of features that are typically associated with noisy and chaotic activity such as balanced excitation and inhibition (C. van Vreeswijk & Sompolinsky, 1996), conductance-based synapses (Cavallari, Panzeri, & Mazzoni, 2014), axonal delays (Cardarilli et al., 2013) and heterogeneity in the neurons parameters can be successfully trained to execute complex temporal tasks.

4.1 Structural perturbation in chaotic systems

Our work on the impact of structural perturbations to chaotic neural networks follows a long line of research on the computational properties of those systems (Bertschinger & Natschläger, 2004; Brunel, 2000; London et al., 2010; Sompolinsky et al., 1988; Sussillo & Abbott, 2009; Sussillo & Barak, 2013; C. van Vreeswijk & Sompolinsky, 1996). Dynamical systems are deemed chaotic based on their response to perturbations on their state. Therefore, most research using chaotic systems in neuroscience has focused on their resistance towards such perturbations. Our work highlights the need to also study the behaviour of such systems when the laws governing the evolution of these states are perturbed. We showed that popular RC implementations that had solutions to dampen the chaotic activity in chaotic networks were extremely vulnerable to very small modifications to their structure. We also showed that not only the magnitude of such perturbation, but also how this perturbation was distributed across the network's structure had considerable impact on how the network behaved. "Global" perturbations, where all connections of a network are randomly changed by a small amount had a much smaller impact on the network dynamics than a "local" perturbation of the same magnitude, but distributed on a few synapses or neurons. Modular reservoirs slightly improved their resilience, but not drastically. One reason why this aspect has been neglected is that chaotic systems that are studied typically have deterministic equations for the evolution of their state-space (Beck & Schögl, 1995). The brain being a biological system, any set of laws governing its evolution are bound to change with time for a multitude of reasons which range from development, neuronal death, neurogenesis or synaptic plasticity. We hope that our

finding will direct more attention toward this aspect of neuronal dynamical systems.

4.2 Neural oscillations

A central finding from this thesis is that neural oscillations can form a powerful code that can control the dynamics of RNNs. This finding is compelling because they are detected across many brain regions in a variety of contexts, from slow waves (<0.05 Hz) to ultra fast (>600 Hz) frequencies (Buzsáki & Draguhn, 2004). Oscillation driven reservoirs become resilient to structural perturbations. Additionally, they are freed from the constraint of using feedback inputs from the readout unit, which also relaxes the constraint of using the large and fast modifications to the weights of the read-out required by FORCE learning.

4.2.1 *Membrane potential oscillations and field oscillations*

There are two main types of oscillatory activity reported in neurophysiological experiments, which are both relevant to our model. Membrane potential oscillations (MPO) are fluctuations of the potential of neurons (Amitai, 1994), and can occur independently in the individual cells of a network (Harvey, Collman, Dombeck, & Tank, 2009). They can be supra-threshold and make the neuron fire action potentials rhythmically (Amir, Michaelis, & Devor, 1999) or remain subthreshold (Schmitz, Gloveli, Behr, Dugladze, & Heinemann, 1998). Such cellular dynamics have been associated with increased reliability and precision of individual action potentials in the presence of noise by quenching the variability in membrane potential (Kuebler et al., 2018; Schaefer, Angelo, Spors, & Margrie, 2006). MPOs can also enhance signal detection in neurons (Engel, Helbig, Russell,

Schimansky-Geier, & Neiman, 2009). The physiological properties underlying these oscillations can be intrinsic to the cells based on their electrophysiological properties (Desmaisons, Vincent, & Lledo, 1999). For instance, sodium conductances (Amir et al., 1999; Desmaisons et al., 1999), calcium channels (S. C. Song, Beatty, & Wilson, 2016), and I_h currents (Lüthi & McCormick, 1998) have been pinpointed as a strong contributor to some MPOs. Although some of these cells might endogenously produce these oscillations (Leung & Yim, 1991), MPOs commonly result from resonance of the neuron. Based on the properties of the ionic conductance of the cells, membrane resonance amplifies only some frequencies present in the input's statistics (Hutcheon & Yarom, 2000; Lampl & Yarom, 1997). Another possible origin for some MPOs is an input drive provided by external stimulation. For instance, (Schaefer et al., 2006) showed that MPOs in mitral cells of the olfactory bulb of mice were locked to their breathing rhythm. However, the most common source identified for MPOs are the synaptic inputs to the oscillating cells. The rhythmic activity of some neuronal populations creates an oscillating synaptic input to individual cells (Buzsáki & Draguhn, 2004; Laurent & Davidowitz, 1994). As we showed in chapter 2, with the right set of parameters, a RNN can generate synchronous events that can lead to oscillations in the potential of the neurons receiving their projections. This leads us to the second main type of oscillatory activity reported in the brain: local field potentials (LFP). LFPs are the extracellular counterparts of MPOs, and are therefore recorded with electrodes capturing the compound activity of multiple neurons (Bremer, 1949). Oscillations in the LFPs are widely studied and a review of their putative functions is beyond this work (see Buzsaki, 2006; Buzsáki & Draguhn, 2004). However, despite much time and resources dedicated to uncovering their function and

origins, we still know very little about them (Wang, 2010). LFPs and MPOs are clearly related to each other, given that concerted synchronized network events have been shown the be at the origin of some MPOs. However, even when they co-occur within a network, MPOs can be decoupled from the LFP (Harvey et al., 2009; Soltesz & Deschenes, 1993). This decoupling often occurs in the form of phase desynchronization, and has been found to be task dependent, where environmental inputs (Harvey et al., 2009) or attentional mechanisms can change the phase of firing of individual cells relative to the LFP. Additionally, the phase on ongoing oscillations might carry information that is complementary to the timing of individual spikes (Kayser, Montemurro, Logothetis, & Panzeri, 2009; Lisman & Buzsáki, 2008). Similarly, a phenomenon that has increasingly been reported in the literature in the last 15 years is cross-frequency coupling (CFC) (Canolty & Knight, 2010; Jensen & Colgin, 2007). The most common form of CFC is the phase-amplitude coupling, where the phase of a slower oscillation typically modulates the amplitude of a faster oscillation (Elswijk et al., 2010; Fries, 2005; Tort et al., 2008). Phase-phase coupling between multiple frequency bands has also been found in many brain regions and context (Belluscio, Mizuseki, Schmidt, Kempter, & Buzsáki, 2012; Jensen & Colgin, 2007). Phase coupling between multiple frequencies suggests that a mechanism regulating the relative phase between different oscillators like suggested by our model is plausible, and used by the brain in some contexts. The functions of phase-phase coupling has mostly been hypothesized to modulate communication between distant brain regions (Darvas, Miller, Rao, & Ojemann, 2009). In this work, we suggest that controlling the phase of oscillations of different frequencies can be used to control the dynamics of RNNs. Overall, it is clear that oscillatory dynamics in neu-

ral circuits are malleable and can be precisely regulated by the brain. Their ubiquity and involvement in many cognitive and behavioural tasks provide compelling support for the oscillatory mechanism that we introduced in this work.

4.2.2 *Multi-periodic codes in the brain*

The strategy that we introduced where multiple oscillators of different periods are combined to create a rich neural code has analogous counterparts used for different models. Fiete et al. (2008) suggested that the irregular spatial periods of the grid cells of the enthorinal cortex (Moser et al., 2008) could serve as a modulo code to encode the exact location of an animal in its environment. In this code analogous to the residue number system (Soderstrand et al., 1986), the remainder of the phases of multiple periodic components represents a unique combination that can be associated to unique locations in the environment. Although deliberations about the model's validity and plausibility are still ongoing (Klukas, Lewis, & Fiete, 2019), the multi-periodic code proposed for the enthorinal grid cells is used specifically to encode spatial location, and isn't formulated in a way that makes it applicable to the temporal domain. There is, however, a similar multi-periodic code that was applied to model temporal representations in a connectionist model. Miall (1989) was the first to suggest that the brain could exploit the mathematical properties of combined oscillators to represent timing intervals. In his model, he showed that combining hundreds of pace-makers with slightly different periods can be used by downstream regions that act as coincidence detectors. This mechanism was reused by Matell & Meck (2004) for the SBF model with a more neurologically grounded framework. In this model, the striatum act as the integrator of the oscillatory activity of multiple cortical

regions, similarly to the read-out unit in our model. However, in order to be efficient, this system requires thousands of oscillators at slightly different frequencies (Matell & Meck, 2004), and they each have to work in a coordinated manner. The strict requirements and the lack of empirical support for such neural activity lead to some scepticism about the theory (Kononowicz & van Wassenhove, 2016). Additionally, this framework can only perform interval timing tasks. Our model, on the other hand, can estimate time intervals lasting several seconds with only two coordinated oscillators, which is several folds lower than the SBF model. Additionally, our model can perform any continuous task, including interval timing.

4.3 Concluding statement

Together, our results suggest that chaotic neural networks are unlikely to autonomously generate complex aperiodic activity. This means that if the brain's neural networks are chaotic, as suggested by some models of the brain, there must be an alternative strategy that can explain how structured patterns of activity can be generated. We propose a hybrid system, where chaotic RNNs aren't the source of such structured activity, but simpler non-chaotic networks projecting to them are. The model uses networks of neurons that oscillates at fixed frequencies to drive chaotic networks and evoke rich patterns of activity. Although driving recurrent network with external inputs has long been proposed as a solution to generate repeatable patterns of activity (Abbott et al., 2016), the source of this input could not be explained without circular reasoning. Indeed, a system that requires a complex time-varying input in order to generate a complex time-varying output leads to little insight into the possible origin of

such neural activity. This model was able to perform a wide range of temporal tasks, including interval timing, drawing 2D shapes, generating natural speech and replaying compressed theta sequences in the hippocampus in a task of spatial navigation. Our work offers a compelling new role for neural oscillations, and provides a powerful mechanism to control the dynamics of spiking RNNs.

5 APPENDIX

Trial-averaging is the standard way to perform analysis on neural data in experimental and theoretical neuroscience (Dayan & Abbott, 2003). This means that the neural activity of a population (usually the activity of isolated cells) is collected across many trials of a given condition, and then averaged across all trials or all presentations of a stimulus. The usual goal is to improve the signal to noise ratio in order to infer the true response of a cell to an experimentally controlled variable. For instance, in a spatial navigation task, the neural activity of the different isolated cells recorded from the hippocampus is usually averaged across all time-points where the mouse is in a given location (Dragoi & Buzsáki, 2006; Fenton et al., 2008; Meshulam, Gauthier, Brody, Tank, & Bialek, 2017). Peaks of average intensity can then be identified and the responding cells are considered as "tuned" to the location. A similar process is made to build tuning curves to model the response of neurons to external stimulations (Butts & Goldman, 2006). This process is also used to model the dynamics of populations of activity during continous tasks such as reaching (Churchland et al., 2012), and extract regularities in fluctuations of their firing rate. Trial-averaged data has also been used in many studies reporting temporal tuning (Baeg et al., 2003; Bakhurin et al., 2017; Bolkan et al., 2017; Fujisawa et al., 2008; Harvey et al., 2012; Heys & Dombeck, 2018; Katlowitz et al., 2018; Mello et al., 2015; Pastalkova et al., 2008; Peters et al., 2014; Runyan et al., 2017; Seidemann et al., 1996; Sheng et al., 2019). Additionally, the average of trial is also done on the convolved activity of the spiking neurons. In some cases, this convolution naturally occurs from the fluorescent dyes that are used to report the neural activity of individual neurons in some experimental settings. Such dyes

are active for periods much longer than individual action potentials. Therefore, the raw data collected by the experimenter is often a distortion of the real spiking activity. But even if the spiking activity could be recorded with infinite temporal resolution, trial-averaging requires some sort of post-hoc convolution or binning. If recorded with infinite precision, no spikes would occur at the same time across trials, so binning procedures or convolutions with non-linear functions are often used. Trial-averaging of convolved data is therefore accepted as an adequate procedure before performing further analysis. In the next sections, we will raise some potential issues that can occur from such methods. Different methodological choices can lead to different results depending on the statistics of the data being analysed. As we show, such distortions can drastically alter the conclusions about the functions of some neurons. We used the results generated in section 3.3.5 on the temporal selectivity of individual neurons to illustrate the impact of these choices. Fig.29a and b show the distribution of each source of input and their derivative for two cells (average across all 30 trials), taken from the network used in Fig. 24d. The top and bottom rows of those panels show the distribution of each type of inputs and derivatives for all the time steps of the simulation (1 second), respectively. The vertical lines on each histogram shows the value at the peak activity. Converting the value of each input statistic at the peak activity to a percentile value, we can see that the first cell (Fig.29a) fires when its maximum combined input is a the extreme of the distribution for the whole trial for both total current and derivative (96th and 92th percentile, respectively). However, the second cell (Fig.29b) scores the middle of the distribution for its total current at peak activity (55th percentile), but scores at the extreme of the distribution for its derivative (99th percentile). This means that the cells is more

sensitive to the derivative of its input than the value of the input for its peak trial-averaged activity. Fig.29c shows the distributions of percentiles for both the total current and derivative for each cell of the network. Overall, the peak activity for each cell is highly influenced by both their total input current and total derivative, as shown by the high percentile of both input statistics around the peak activity. The value of both types of input statistics at the peak activity are uncorrelated with each other (Fig.29d, Pearson $r = 0.03$, $p = 0.48$). However, when using a larger Gaussian kernel to filter the spiking data, the impact of the derivative disappears (Fig.29e), as shown from the normally distributed derivative around the time of peak activity. Under these conditions, the only input statistic relevant to the peak activity is the combined current, and both input statistics are also uncorrelated (Fig.29f, Pearson $r = -0.07$, $p = 0.1$). To better capture the impact of this convolution kernel, we extracted the average input and the average derivative of the input received at the trial averaged peak times by individual cells in the network (see section 3.3.5 in chapter 2 for full implementation). The impact of the convolution kernel was apparent in aligned of the currents with the timing of peak activity. Fig. 30a shows an offset between the peak current receive by the network's neurons and their peak trial-averaged activity. There was a perfect alignment with the peak derivative of their input and their peak trial-averaged activity. With a slower convolution kernel, the peak current was aligned with the timing of the trial-averaged peak, and the effect of the derivative vanished. Fig. 30b shows that the type of activity behind trial-averaged peaks changed with different convolution kernels. Faster kernels lead to the detection of precise and reliable spikes, whereas slower kernels biased the detection towards timings where the individual trials had higher firing-rates (Fig.31). Thus, precau-

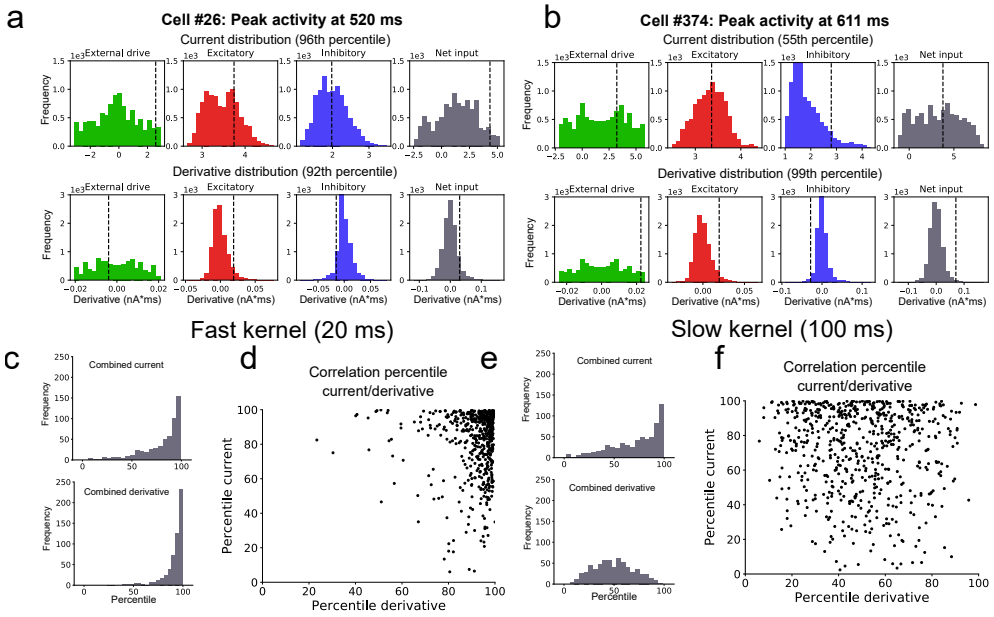

Figure 29: Dynamics underlying trial-averaged temporal tuning. **a** and **b** Distributions of the trial-averaged sources of inputs for two cells of the network. The top row of each panel shows the distribution of the input at each time step, and the bottom row shows the derivative of the input at each time step. **c** Distribution of the percentile of the total combined currents (top) and (bottom) combined derivatives for each cell at the time of the peak in the trial-averaged activity. **d** Correlation between the percentile of the total current and derivative received by each cell at the time of the trial-averaged peak. **e,f** Same as **c,d**, but with a slow kernel.

tions should be taken when interpreting results from trial-averaged data. To get a better intuition behind the influence of the Gaussian kernel on the detection of temporally locked peak activity, we developed a simplified model of temporal tuning. We define two types of peak activity: global and local maximum. A local maximum refers to the peak activity recorded on an individual trial. Global maxima refer to peak in the trial-averaged activity. We simulated two modes of activity and we used two types of kernel (fast: s.d. = 5 ms, slow: s.d. = 50 ms) to detect the peak activity. For the first condition, we simulated 10 Poisson spike trains with a firing rate of 10 Hz for one second, and we added 1 spike half-way through each trial at 500 ms, with a Gaussian jitter of 3 ms (Fig.30c (top)). This simulates 10 trials

of the activity of one cell that is temporally tuned to a 500 ms interval. The fast kernel adequately identifies the global peak in activity at 500 ms, as shown on the peak average of the convolved traces. However, we can see that the time of the trial-averaged peak does not correlate with the local peaks (denoted by the blue stars) in activity on most trials. This means that in this scenario, neurons don't increase their firing rate at the time of the trial-averaged peak, and that a precise and reliable spike leads to the global peak. The opposite is true for a slower kernel, which is more sensitive to a local increase in firing rate rather than precise and reliable spikes. A slower kernel used on the same raster detects a global peak that isn't centered at the time of the additional spike at 500 ms due to sensitivity to local increase in firing rate that matches the time scale of the kernel. In a condition where more than one spike is added at the preferred interval, a slower kernel is more adequate for global peak detection. In Fig.30c (bottom), six spikes were added with a Gaussian distribution centered at 500 ms, with a s.d. of 60 ms. This results in an elevated firing rate on every trial around the center of the distribution as identified with the slow kernel. The fast kernel performs poorly for the identification of the local peak activity on a trial-to-trial basis, and can also lead to aberrant global peak detection. We therefore tested if these two kernel sizes would lead to the detection of different types of activity (local increase v. reliable and precise activity) in our data. We calculated an index of local activity, by extracting the firing rates of each neuron at each time step, and obtained the percentile of this firing rate at the trial-averaged peak for each cell (Fig.30d). The distribution of trial-averaged peaks of the fast kernel was shifted considerably to the left compared to the slow kernel. In other words, slower kernels are biased toward the detection of increases in local firing rate for each

cell, whereas the fast kernels are biased towards precise and reliable spikes that do not necessarily coincide with local peaks in activity. Therefore, the choice of the kernel has a considerable impact on the type of activity underlying the trial-averaged peaks.

This difference between "precise and reliable" and "local increase in firing rate" activity is a product of the amplitude of the peak activity, the variability of the added spikes and the width of the convolution kernel. Therefore, the type of dynamics detected with trial-averaged activity arise from the interaction between the kinetics of the firing rate fluctuations and the kinetics of the convolution filter. We performed additional simulations to isolate the contribution of each of these factors on the type of dynamics underlying the trial-averaged peaks (Fig.31). In this analysis, we added time-locked spikes at 1 seconds on trials lasting 2 seconds, superposed with multiple trials with Poisson activity. We systematically varied the jitter of time-locked spikes (sigma peak), the number of added spikes, the Poisson activity rate and the kinetics of the convolution kernel (sigma filter). We then looked at the precision of the global peak (maximal activity of the trial-averaged data), and the percentile of the activity at the time of the global peak (this is an indicator of "local" activity: higher percentile means that the neuron was more active at the timing of the global peak than all other time steps of the trial). The results confirmed what we had observed in Fig. 30. When the jitter of additional spikes is large and there are more added spikes, slower kernels tend to perform better. However, when only one spike is added, faster kernels perform better across a wider range of conditions. Therefore, there are many conditions where the local activity is low at the timing of the global peak, and where global peaks do not correspond to local peaks in activity.

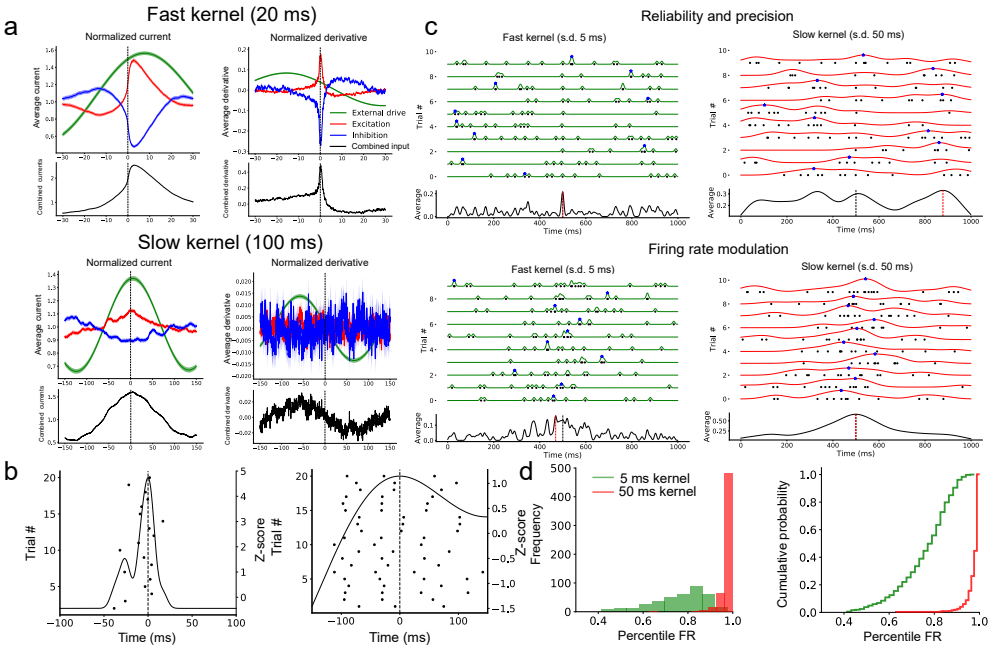

Figure 30: Modelling dynamics of trial-averaged peaks. **a** Normalised average excitatory, inhibitory and external input received by each cell aligned on their peak firing rate (at t = 0 ms), as detected with a fast kernel (s.d. = 20 ms, top) and a slow kernel (s.d. = 100 ms, bottom). The left-hand side shows the average current whereas the right-hand side shows the average derivative of the same currents around the peak trial-averaged activity. The combined inputs (external drive + excitation - inhibition) is represented on the bottom. **b** Trial-averaged peak for the same cell as detected by a fast and a slow kernel. With the fast kernel (left), a single spike per trial is responsible for the trial-averaged peak in activity. With the larger kernel however (right), multiple spikes per trial are responsible for the peak activity detected across trials. **c** Top: Spike raster generated from a Poisson spike train at 10 Hz with one spike with a timing drawn from a normal distribution $\mathrm{N}(500, 3)$ ms. The left figure shows the convolution of each trial, as well as the trial-averaged convolution (bottom trace) with a fast kernel, and the right figure shows the same raster convolved with a slow kernel. The blue stars shows the peak for each trial based on the maximum value of the convolution. The vertical dotted black line shows the 500 ms center of expected peak, and the vertical red dotted line shows the detected peak based on the maximum of the trial-averaged convolution. Bottom: Six spikes with a timing drawn from a normal distribution $\mathrm{N}(500, 50)$ ms. **d** Left: Histogram of the percentile of the within-trial activity at the trial-averaged peak, as detected with the fast and the slow kernels. Right: cumulative distribution function of the percentiles as detected with the fast and the slow kernels.

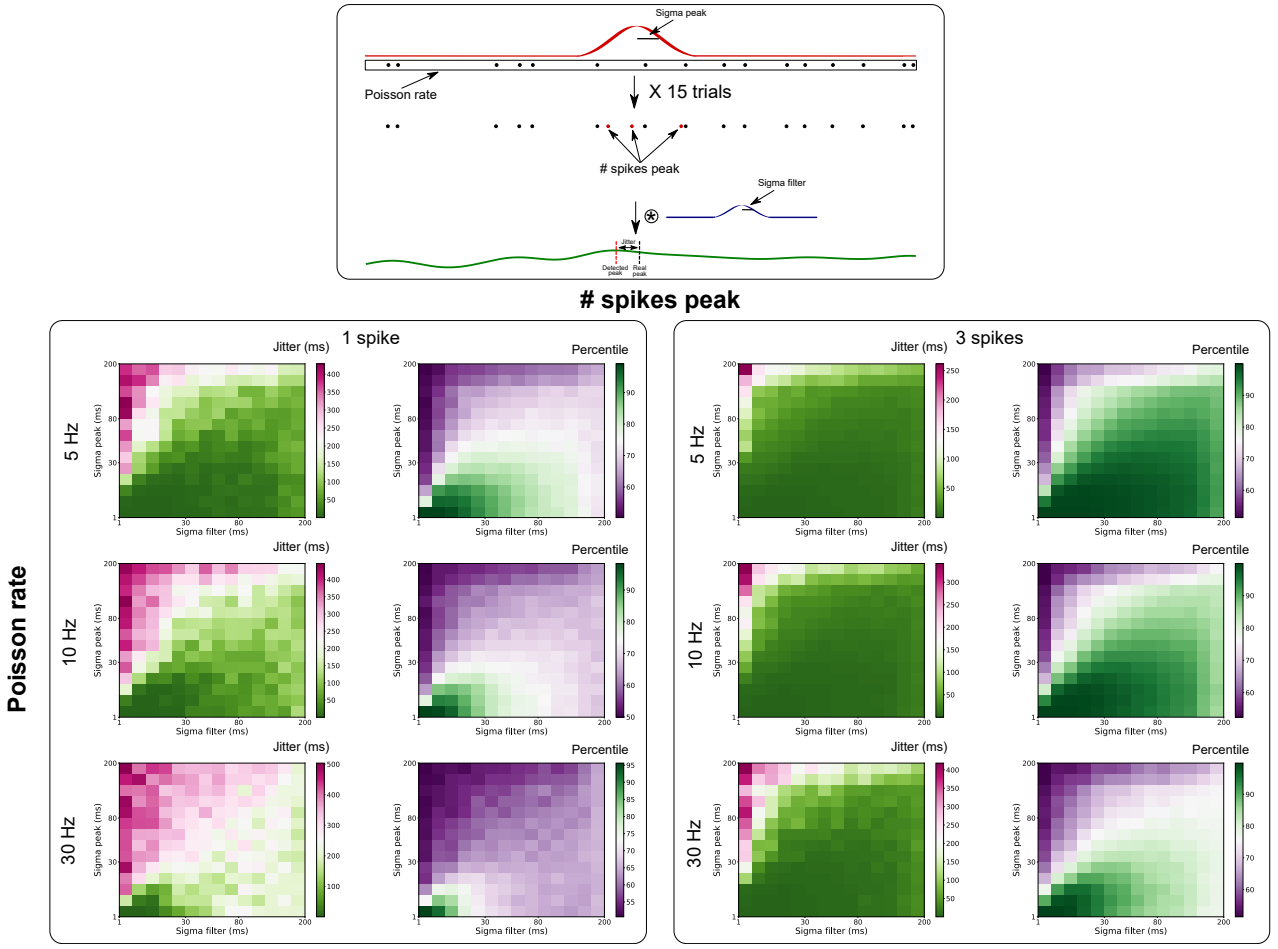

Figure 31: Detection of peak activity with trial-averaged data. We simulated different conditions to investigate the type of dynamics underlying trial-average peaks. The top schema illustrates the different parameters that we explored. For each condition, we generated 15 Poissons spike trains (*trials*) with different firing rates (5, 10 and 30 Hz). We then added different numbers of spikes (# *spikes peak*) on top of the Poisson trains, with a normal distribution with a mean = 1 second (each trial is 2 seconds), s.d. = *sigma peak*. We then convolved each spike train with a Gaussian filter with a s.d. = *sigma filter*. Each condition is repeated 100 times, and we reported the average *jitter* (i.e. error) and *percentile* (higher percentile values means that there is a better correspondence between local and global peaks in activity).

REFERENCES

Abarbanel, H. D., Creveling, D. R., & Jeanne, J. M. (2008). Estimation of parameters in nonlinear systems using balanced synchronization. *Physical Review E*, *77*(1), 016208.

Abbott, L. F., DePasquale, B., & Memmesheimer, R.-M. (2016, February). Building functional networks of spiking model neurons. *Nature Neuroscience*, *19*, 350. Retrieved from https://doi.org/10.1038/nn.4241

Abeles, M. (1982). *Local cortical circuits: an electrophysiological study*. Berlin; New York: Springer-Verlag. Retrieved from http://dx.doi.org/10.1007/978-3-642-81708-3

Akam, T., Oren, I., Mantoan, L., Ferenczi, E., & Kullmann, D. M. (2012). Oscillatory dynamics in the hippocampus support dentate gyrus-CA3 coupling. *Nature neuroscience*, *15*(5), 763–768.

Alluri, R. K., Rose, G. J., Hanson, J. L., Leary, C. J., Vasquez-Opazo, G. A., Graham, J. A., & Wilkerson, J. (2016, March). Phasic, suprathreshold excitation and sustained inhibition underlie neuronal selectivity for short-duration sounds. *Proceedings of the National Academy of Sciences*, *113*(13), E1927–E1935. Retrieved 2019-12-11, from https://www.pnas.org/content/113/13/E1927 doi: 10.1073/pnas.1520971113

Amir, R., Michaelis, M., & Devor, M. (1999). Membrane potential oscillations in dorsal root ganglion neurons: role in normal electrogenesis and neuropathic pain. *The journal of Neuroscience*, *19*(19), 8589–8596.

Amit, D. J., & Amit, D. J. (1992). *Modeling brain function: The world of attractor neural networks*. Cambridge university press.

Amit, D. J., & Brunel, N. (1997). Model of global spontaneous activity

and local structured activity during delay periods in the cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, *7*(3), 237–252.

Amitai, Y. (1994, November). Membrane potential oscillations underlying firing patterns in neocortical neurons. *Neuroscience*, *63*(1), 151–161. Retrieved 2019-08-21, from http://www.sciencedirect.com/science/article/pii/0306452294900132 doi: 10.1016/0306-4522(94)90013-2

Anumanchipalli, G. K., Chartier, J., & Chang, E. F. (2019, April). Speech synthesis from neural decoding of spoken sentences. *Nature*, *568*(7753), 493. Retrieved 2019-07-26, from https://www.nature.com/articles/s41586-019-1119-1 doi: 10.1038/s41586-019-1119-1

Arieli, A., Sterkin, A., Grinvald, A., & Aertsen, A. (1996). Dynamics of ongoing activity: explanation of the large variability in evoked cortical responses. *Science*, *273*(5283), 1868–1871.

Baeg, E. H., Kim, Y. B., Huh, K., Mook-Jung, I., Kim, H. T., & Jung, M. W. (2003, September). Dynamics of Population Code for Working Memory in the Prefrontal Cortex. *Neuron*, *40*(1), 177–188. Retrieved 2019-08-14, from http://www.sciencedirect.com/science/article/pii/S089662730300597X doi: 10.1016/S0896-6273(03)00597-X

Bai, Z. (1997). Circular law. *The Annals of Probability*, *25*(1), 494–529.

Bakhurin, K. I., Goudar, V., Shobe, J. L., Claar, L. D., Buonomano, D. V., & Masmanidis, S. C. (2017, January). Differential Encoding of Time by Prefrontal and Striatal Network Dynamics. *The Journal of Neuroscience*, *37*(4), 854. Retrieved from http://www.jneurosci.org/content/37/4/854.abstract doi: 10.1523/JNEUROSCI.1789-16.2017

Balci, F., Ludvig, E. A., Abner, R., Zhuang, X., Poon, P., &

Brunner, D. (2010, April). Motivational effects on interval timing in dopamine transporter (DAT) knockdown mice. *Brain Research*, *1325*, 89–99. Retrieved 2019-08-06, from http://www.sciencedirect.com/science/article/pii/S0006899310003811 doi: 10.1016/j.brainres.2010.02.034

Banerjee, A., Seriès, P., & Pouget, A. (2007, December). Dynamical Constraints on Using Precise Spike Timing to Compute in Recurrent Cortical Networks. *Neural Computation*, *20*(4), 974–993. Retrieved 2019-07-28, from https://doi.org/10.1162/neco.2008.05-06-206 doi: 10.1162/neco.2008.05-06-206

Barak, O., Sussillo, D., Romo, R., Tsodyks, M., & Abbott, L. (2013, April). From fixed points to chaos: Three models of delayed discrimination. *Conversion of Sensory Signals into Perceptions, Memories and Decisions*, *103*(0), 214–222. Retrieved from http://www.sciencedirect.com/science/article/pii/S0301008213000129 doi: 10.1016/j.pneurobio.2013.02.002

Beck, C., & Schögl, F. (1995). *Thermodynamics of chaotic systems: an introduction*. Cambridge University Press.

Belluscio, M. A., Mizuseki, K., Schmidt, R., Kempter, R., & Buzsáki, G. (2012, January). Cross-Frequency Phase–Phase Coupling between Theta and Gamma Oscillations in the Hippocampus. *Journal of Neuroscience*, *32*(2), 423–435. Retrieved 2019-08-22, from https://www.jneurosci.org/content/32/2/423 doi: 10.1523/JNEUROSCI.4122-11.2012

Bengio, Y., Lee, D.-H., Bornschein, J., Mesnard, T., & Lin, Z. (2015, February). Towards Biologically Plausible Deep Learning. *arXiv:1502.04156 [cs]*. Retrieved 2019-08-03, from http://arxiv.org/abs/1502.04156 (arXiv: 1502.04156)

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, *5*(2), 157–166.

Berberian, N., MacPherson, A., Giraud, E., Richardson, L., & Thivierge, J.-P. (2017, February). Neuronal pattern separation of motion-relevant input in LIP activity. *Journal of Neurophysiology*, *117*(2), 738–755. Retrieved 2019-08-14, from http://www.physiology.org/doi/10.1152/jn.00145.2016 doi: 10.1152/jn.00145.2016

Bernacchia, A., Seo, H., Lee, D., & Wang, X.-J. (2011, March). A reservoir of time constants for memory traces in cortical neurons. *Nat Neurosci*, *14*(3), 366–372. Retrieved from http://dx.doi.org/10.1038/nn.2752 doi: 10.1038/nn.2752

Bertschinger, N., & Natschläger, T. (2004). Real-time computation at the edge of chaos in recurrent neural networks. *Neural computation*, *16*(7), 1413–1436.

Bi, G.-q., & Poo, M.-m. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *The Journal of neuroscience*, *18*(24), 10464–10472.

Bittner, K. C., Milstein, A. D., Grienberger, C., Romani, S., & Magee, J. C. (2017, September). Behavioral time scale synaptic plasticity underlies CA1 place fields. *Science*, *357*(6355), 1033–1036. Retrieved 2019-06-26, from https://science.sciencemag.org/content/357/6355/1033 doi: 10.1126/science.aan3846

Bolkan, S. S., Stujenske, J. M., Parnaudeau, S., Spellman, T. J., Rauffenbart, C., Abbas, A. I., ... Kellendonk, C. (2017). Thalamic projections sustain prefrontal activity during working memory maintenance. *Nature neuroscience*, *20*(7), 987.

Bonifazi, P., Goldin, M., Picardo, M. A., Jorquera, I., Cattani, A., Bianconi, G., ... Cossart, R. (2009). GABAergic hub neurons orchestrate synchrony in developing hippocampal networks. *Science*, *326*(5958), 1419–1424.

Bremer, F. (1949, January). Considérations sur l'origine et la nature des ondes cérébrales. *Electroencephalography and Clinical Neurophysiology*, *1*(1), 177–193. Retrieved 2019-08-21, from http://www.sciencedirect.com/science/article/pii/0013469449901741 doi: 10.1016/0013-4694(49)90174-1

Brin, M., & Stuck, G. (2002). *Introduction to dynamical systems*. Cambridge University Press.

Brody, C. D., Hernández, A., Zainos, A., & Romo, R. (2003, November). Timing and Neural Encoding of Somatosensory Parametric Working Memory in Macaque Prefrontal Cortex. *Cerebral Cortex*, *13*(11), 1196–1207. Retrieved 2019-08-14, from https://academic.oup.com/cercor/article/13/11/1196/274114 doi: 10.1093/cercor/bhg100

Brody, C. D., Romo, R., & Kepecs, A. (2003, April). Basic mechanisms for graded persistent activity: discrete attractors, continuous attractors, and dynamic representations. *Current Opinion in Neurobiology*, *13*(2), 204–211. Retrieved 2019-08-02, from http://www.sciencedirect.com/science/article/pii/S0959438803000503 doi: 10.1016/S0959-4388(03)00050-3

Broome, B. M., Jayaraman, V., & Laurent, G. (2006, August). Encoding and Decoding of Overlapping Odor Sequences. *Neuron*, *51*(4), 467–482. Retrieved from http://www.sciencedirect.com/science/article/pii/S0896627306005587 doi: 10.1016/j.neuron.2006.07.018

Brosch, M., Selezneva, E., & Scheich, H. (2005, July). Nonauditory

Events of a Behavioral Procedure Activate Auditory Cortex of Highly Trained Monkeys. *Journal of Neuroscience*, *25*(29), 6797–6806. Retrieved 2019-08-07, from https://www.jneurosci.org/content/25/29/6797 doi: 10.1523/JNEUROSCI.1571-05.2005

Brunel, N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of computational neuroscience*, *8*(3), 183–208.

Brunel, N., & Hansel, D. (2006, March). How Noise Affects the Synchronization Properties of Recurrent Networks of Inhibitory Neurons. *Neural Computation*, *18*(5), 1066–1110. Retrieved 2019-08-20, from https://doi.org/10.1162/neco.2006.18.5.1066 doi: 10.1162/neco.2006.18.5.1066

Bueti, D., Bahrami, B., & Walsh, V. (2008). Sensory and association cortex in time perception. *Journal of Cognitive Neuroscience*, *20*(6), 1054–1062.

Buhusi, C. V., & Meck, W. H. (2002). Differential effects of methamphetamine and haloperidol on the control of an internal clock. *Behavioral Neuroscience*, *116*(2), 291–297. doi: 10.1037/0735-7044.116.2.291

Buhusi, C. V., & Meck, W. H. (2005, October). What makes us tick? Functional and neural mechanisms of interval timing. *Nat Rev Neurosci*, *6*(10), 755–765. Retrieved from http://dx.doi.org/10.1038/nrn1764 doi: 10.1038/nrn1764

Buhusi, C. V., & Meck, W. H. (2009). Relativity theory and time perception: single or multiple clocks? *PloS one*, *4*(7), e6268.

Buonomano, D. V. (2000, February). Decoding Temporal Information: A Model Based on Short-Term Synaptic Plasticity. *Journal of Neuroscience*, *20*(3), 1129–1141. Retrieved 2019-08-02, from https://www.jneurosci.org/content/20/3/1129 doi: 10.1523/

JNEUROSCI.20-03-01129.2000

Buonomano, D. V. (2003). Timing of neural responses in cortical organotypic slices. *Proceedings of the National Academy of Sciences*, *100*(8), 4897–4902.

Buonomano, D. V., & Maass, W. (2009). State-dependent computations: spatiotemporal processing in cortical networks. *Nature Reviews Neuroscience*, *10*(2), 113–125.

Buonomano, D. V., & Merzenich, M. M. (1995). Temporal information transformed into a spatial code by a neural network with realistic properties. *Science*, *267*(5200), 1028–1030.

Burgess, N., Barry, C., & O'Keefe, J. (2007). An oscillatory interference model of grid cell firing. *Hippocampus*, *17*(9), 801–812. Retrieved 2019-08-08, from https://onlinelibrary.wiley.com/doi/abs/10.1002/hipo.20327 doi: 10.1002/hipo.20327

Burr, D., Tozzi, A., & Morrone, M. C. (2007, April). Neural mechanisms for timing visual events are spatially selective in real-world coordinates. *Nature Neuroscience*, *10*(4), 423–425. Retrieved 2019-08-01, from http://www.nature.com/articles/nn1874 doi: 10.1038/nn1874

Butts, D. A., & Goldman, M. S. (2006, March). Tuning Curves, Neuronal Variability, and Sensory Coding. *PLOS Biology*, *4*(4), e92. Retrieved 2019-08-23, from https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.0040092 doi: 10.1371/journal.pbio.0040092

Buxhoeveden, D. P., & Casanova, M. F. (2002). The minicolumn hypothesis in neuroscience. *Brain*, *125*(5), 935–951.

Buzsaki, G. (2006). *Rhythms of the Brain*. Oxford University Press.

Buzsáki, G., & Draguhn, A. (2004). Neuronal oscillations in cortical networks. *science*, *304*(5679), 1926–1929.

Canolty, R. T., & Knight, R. T. (2010). The functional role of cross-frequency coupling. *Trends in cognitive sciences*, *14*(11), 506–515.

Cardarilli, G. C., Cristini, A., Nunzio, L. D., Re, M., Salerno, M., & Susi, G. (2013, November). Spiking neural networks based on LIF with latency: Simulation and synchronization effects. In *2013 Asilomar Conference on Signals, Systems and Computers* (pp. 1838–1842). doi: 10.1109/ACSSC.2013.6810620

Carr, C., & Konishi, M. (1990, October). A circuit for detection of interaural time differences in the brain stem of the barn owl. *The Journal of Neuroscience*, *10*(10), 3227–3246. Retrieved 2019-05-21, from http://www.jneurosci.org/lookup/doi/10.1523/JNEUROSCI.10-10-03227.1990 doi: 10.1523/JNEUROSCI.10-10-03227.1990

Carr, C. E., & Konishi, M. (1988, November). Axonal delay lines for time measurement in the owl's brainstem. *Proceedings of the National Academy of Sciences*, *85*(21), 8311–8315. Retrieved 2019-08-01, from https://www.pnas.org/content/85/21/8311 doi: 10.1073/pnas.85.21.8311

Cavallari, S., Panzeri, S., & Mazzoni, A. (2014). Comparison of the dynamics of neural interactions between current-based and conductance-based integrate-and-fire recurrent networks. *Frontiers in Neural Circuits*, *8*. Retrieved 2019-08-16, from https://www.frontiersin.org/articles/10.3389/fncir.2014.00012/full doi: 10.3389/fncir.2014.00012

Chartier, S., & Boukadoum, M. (2006, March). A bidirectional heteroassociative memory for binary and grey-level patterns. *IEEE Transactions on Neural Networks*, *17*(2), 385–396. doi: 10.1109/TNN.2005.863420

Chartier, S., & Boukadoum, M. (2011). *Encoding Static and Tempo-*

ral Patterns with a Bidirectional Heteroassociative Memory [Research article]. Retrieved 2019-12-11, from https://www.hindawi.com/journals/jam/2011/301204/ doi: 10.1155/2011/301204

Chiba, A., Oshio, K.-i., & Inase, M. (2015). Neuronal representation of duration discrimination in the monkey striatum. *Physiological Reports*, *3*(2), e12283. Retrieved 2019-08-07, from https://physoc.onlinelibrary.wiley.com/doi/abs/10.14814/phy2.12283 doi: 10.14814/phy2.12283

Chubykin, A., Roach, E., Bear, M., & Shuler, M. (2013, February). A Cholinergic Mechanism for Reward Timing within Primary Visual Cortex. *Neuron*, *77*(4), 723–735. Retrieved from http://www.sciencedirect.com/science/article/pii/S0896627313000470 doi: 10.1016/j.neuron.2012.12.039

Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Foster, J. D., Nuyujukian, P., Ryu, S. I., & Shenoy, K. V. (2012, July). Neural population dynamics during reaching. *Nature*, *487*(7405), 51–56. Retrieved from http://dx.doi.org/10.1038/nature11129 doi: 10.1038/nature11129

Churchland, M. M., & Shenoy, K. V. (2007). Temporal complexity and heterogeneity of single-neuron activity in premotor and motor cortex. *Journal of neurophysiology*, *97*(6), 4235–4257. doi: 10.1152/jn.00095.2007

Conant, D. F., Bouchard, K. E., Leonard, M. K., & Chang, E. F. (2018, March). Human Sensorimotor Cortex Control of Directly Measured Vocal Tract Movements during Vowel Production. *Journal of Neuroscience*, *38*(12), 2955–2966. Retrieved 2019-07-26, from https://www.jneurosci.org/content/38/12/2955 doi: 10.1523/JNEUROSCI.2382-17.2018

Constantinidis, C., Franowicz, M. N., & Goldman-Rakic, P. S. (2001, May). Coding Specificity in Cortical Microcircuits: A Multiple-Electrode Analysis of Primate Prefrontal Cortex. *Journal of Neuroscience*, *21*(10), 3646–3655. Retrieved 2019-08-14, from https://www.jneurosci.org/content/21/10/3646 doi: 10.1523/JNEUROSCI.21-10-03646.2001

Coull, J. T., Cheng, R.-K., & Meck, W. H. (2011, January). Neuroanatomical and Neurochemical Substrates of Timing. *Neuropsychopharmacology*, *36*(1), 3–25. Retrieved 2019-08-06, from https://www.nature.com/articles/npp2010113 doi: 10.1038/npp.2010.113

Darlington, T. K., Wager-Smith, K., Ceriani, M. F., Staknis, D., Gekakis, N., Steeves, T. D., ... Kay, S. A. (1998). Closing the circadian loop: CLOCK-induced transcription of its own inhibitors per and tim. *Science*, *280*(5369), 1599–1603.

Darvas, F., Miller, K. J., Rao, R. P. N., & Ojemann, J. G. (2009, January). Nonlinear Phase–Phase Cross-Frequency Coupling Mediates Communication between Distant Sites in Human Neocortex. *Journal of Neuroscience*, *29*(2), 426–435. Retrieved 2019-08-22, from https://www.jneurosci.org/content/29/2/426 doi: 10.1523/JNEUROSCI.3688-08.2009

Dayan, P., & Abbott, L. (2003). Theoretical neuroscience: computational and mathematical modeling of neural systems. *Journal of Cognitive Neuroscience*, *15*(1), 154–155.

Desmaisons, D., Vincent, J.-D., & Lledo, P.-M. (1999). Control of action potential timing by intrinsic subthreshold oscillations in olfactory bulb output neurons. *The Journal of Neuroscience*, *19*(24), 10727–10737.

Destexhe, A. (1997). Conductance-based integrate-and-fire models.

Neural Computation, *9*(3), 503–514.

Deuchars, J., & Thomson, A. M. (1996, October). CA1 pyramid-pyramid connections in rat hippocampus in vitro: Dual intracellular recordings with biocytin filling. *Neuroscience*, *74*(4), 1009–1018. Retrieved 2019-11-28, from http://www.sciencedirect.com/science/article/pii/0306452296002515 doi: 10.1016/0306-4522(96)00251-5

De Zeeuw, C. I., Hoebeek, F. E., Bosman, L. W., Schonewille, M., Witter, L., & Koekkoek, S. K. (2011). Spatiotemporal firing patterns in the cerebellum. *Nature Reviews Neuroscience*, *12*(6), 327–344.

Diesmann, M., Gewaltig, M.-O., & Aertsen, A. (1999). Stable propagation of synchronous spiking in cortical neural networks. *Nature*, *402*(6761), 529–533.

Doya, K. (2000). Complementary roles of basal ganglia and cerebellum in learning and motor control. *Current opinion in neurobiology*, *10*(6), 732–739.

Dragoi, G., & Buzsáki, G. (2006, April). Temporal Encoding of Place Sequences by Hippocampal Cell Assemblies. *Neuron*, *50*(1), 145–157. Retrieved 2019-07-25, from http://www.sciencedirect.com/science/article/pii/S0896627306001656 doi: 10.1016/j.neuron.2006.02.023

Dragoi, G., Carpi, D., Recce, M., Csicsvari, J., & Buzsáki, G. (1999, July). Interactions between Hippocampus and Medial Septum during Sharp Waves and Theta Oscillation in the Behaving Rat. *Journal of Neuroscience*, *19*(14), 6191–6199. Retrieved 2019-07-22, from https://www.jneurosci.org/content/19/14/6191 doi: 10.1523/JNEUROSCI.19-14-06191.1999

Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern classification*. John Wiley & Sons.

Elswijk, G. v., Maij, F., Schoffelen, J.-M., Overeem, S., Stegeman, D. F., & Fries, P. (2010, March). Corticospinal Beta-Band Synchronization Entails Rhythmic Gain Modulation. *Journal of Neuroscience*, *30*(12), 4481–4488. Retrieved 2019-08-22, from https://www.jneurosci.org/content/30/12/4481 doi: 10.1523/JNEUROSCI.2794-09.2010

Enel, P. (2014). *Dynamic representation in the prefrontal cortex : insights from comparing reservoir computing and primate neurophysiology* (Theses, Université Claude Bernard - Lyon I). Retrieved from https://tel.archives-ouvertes.fr/tel-01056696

Engel, T. A., Helbig, B., Russell, D. F., Schimansky-Geier, L., & Neiman, A. B. (2009). Coherent stochastic oscillations enhance signal detection in spiking neurons. *Physical Review E*, *80*(2), 021919.

Engineer, C. T., Perez, C. A., Chen, Y. H., Carraway, R. S., Reed, A. C., Shetake, J. A., ... Kilgard, M. P. (2008). Cortical activity patterns predict speech discrimination ability. *Nature neuroscience*, *11*(5), 603–608.

Ermentrout, B. (1996). Type I membranes, phase resetting curves, and synchrony. *Neural computation*, *8*(5), 979–1001.

Farries, M. A., & Wilson, C. J. (2012, October). Phase response curves of subthalamic neurons measured with synaptic input and current injection. *Journal of Neurophysiology*, *108*(7), 1822–1837. Retrieved from http://jn.physiology.org/content/108/7/1822.abstract doi: 10.1152/jn.00053.2012

Fenton, A. A., Kao, H.-Y., Neymotin, S. A., Olypher, A., Vayntrub, Y., Lytton, W. W., & Ludvig, N. (2008, October). Unmasking the CA1 Ensemble Place Code by Exposures to Small and Large Environments: More Place Cells and Multiple, Irregularly Ar-

ranged, and Expanded Place Fields in the Larger Space. *Journal of Neuroscience*, *28*(44), 11250–11262. Retrieved 2019-08-23, from https://www.jneurosci.org/content/28/44/11250 doi: 10.1523/JNEUROSCI.2862-08.2008

Fiete, I. R., Burak, Y., & Brookings, T. (2008). What grid cells convey about rat location. *The Journal of Neuroscience*, *28*(27), 6858–6871.

Fiorillo, C. D., Newsome, W. T., & Schultz, W. (2008, August). The temporal precision of reward prediction in dopamine neurons. *Nature Neuroscience*, *11*(8), 966–973. Retrieved 2019-08-06, from https://www.nature.com/articles/nn.2159 doi: 10.1038/nn.2159

Folgheraiter, M., Keldibek, A., Aubakir, B., Gini, G., Franchi, A. M., & Bana, M. (2019, July). A neuromorphic control architecture for a biped robot. *Robotics and Autonomous Systems*, S0921889017301793. Retrieved 2019-07-27, from https://linkinghub.elsevier.com/retrieve/pii/S0921889017301793 doi: 10.1016/j.robot.2019.07.014

Foster, D. J., & Wilson, M. A. (2006, March). Reverse replay of behavioural sequences in hippocampal place cells during the awake state. *Nature*, *440*(7084), 680. Retrieved 2019-07-25, from https://www.nature.com/articles/nature04587 doi: 10.1038/nature04587

Foster, D. J., & Wilson, M. A. (2007). Hippocampal theta sequences. *Hippocampus*, *17*(11), 1093–1099. Retrieved 2019-07-26, from https://onlinelibrary.wiley.com/doi/abs/10.1002/hipo.20345 doi: 10.1002/hipo.20345

Frank, L. M., Brown, E. N., & Wilson, M. (2000, July). Trajectory Encoding in the Hippocampus and Entorhinal Cortex. *Neuron*, *27*(1), 169–178. Retrieved 2019-

07-25, from http://www.sciencedirect.com/science/article/pii/S0896627300000180 doi: 10.1016/S0896-6273(00)00018-0

Friedrich, R. W., & Laurent, G. (2001, February). Dynamic Optimization of Odor Representations by Slow Temporal Patterning of Mitral Cell Activity. *Science*, *291*(5505), 889–894. Retrieved 2019-08-14, from https://science.sciencemag.org/content/291/5505/889 doi: 10.1126/science.291.5505.889

Fries, P. (2005, October). A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. *Trends in Cognitive Sciences*, *9*(10), 474–480. Retrieved from http://www.sciencedirect.com/science/article/pii/S1364661305002421 doi: 10.1016/j.tics.2005.08.011

Froemke, R. C., & Schreiner, C. E. (2015). Synaptic plasticity as a cortical coding scheme. *Current opinion in neurobiology*, *35*, 185–199.

Fu, Y., Yu, Y., Paxinos, G., Watson, C., & Rusznák, Z. (2015, April). Aging-dependent changes in the cellular composition of the mouse brain and spinal cord. *Neuroscience*, *290*(0), 406–420. Retrieved from http://www.sciencedirect.com/science/article/pii/S0306452215000949 doi: 10.1016/j.neuroscience.2015.01.039

Fujisawa, S., Amarasingham, A., Harrison, M. T., & Buzsáki, G. (2008, July). Behavior-dependent short-term assembly dynamics in the medial prefrontal cortex. *Nature Neuroscience*, *11*(7), 823–833. Retrieved 2019-08-14, from https://www.nature.com/articles/nn.2134 doi: 10.1038/nn.2134

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

Retrieved 2019-08-14, from https://www.physiology.org/doi/abs/10.1152/jn.1989.61.2.331 doi: 10.1152/jn.1989.61.2.331

Fuster, J. M. (1973, January). Unit activity in prefrontal cortex during delayed-response performance: neuronal correlates of transient memory. *Journal of Neurophysiology*, *36*(1), 61–78. Retrieved 2019-08-13, from https://www.physiology.org/doi/abs/10.1152/jn.1973.36.1.61 doi: 10.1152/jn.1973.36.1.61

Fuster, J. M., & Alexander, G. E. (1971, August). Neuron Activity Related to Short-Term Memory. *Science*, *173*(3997), 652. Retrieved from http://science.sciencemag.org/content/173/3997/652.abstract doi: 10.1126/science.173.3997.652

Fuster, J. M., & Alexander, G. E. (1973, October). Firing changes in cells of the nucleus medialis dorsalis associated with delayed response behavior. *Brain Research*, *61*, 79–91. Retrieved 2019-08-13, from http://www.sciencedirect.com/science/article/pii/0006899373905179 doi: 10.1016/0006-8993(73)90517-9

Gallicchio, C., Micheli, A., & Pedrelli, L. (2017, December). Deep reservoir computing: A critical experimental analysis. *Neurocomputing*, *268*, 87–99. Retrieved 2019-07-27, from https://linkinghub.elsevier.com/retrieve/pii/S0925231217307567 doi: 10.1016/j.neucom.2016.12.089

Gerstner, W., Kempter, R., van Hemmen, J. L., & Wagner, H. (1996). A neuronal learning rule for sub-millisecond temporal coding. *Nature*, *383*(6595), 76.

Gerstner, W., Lehmann, M., Liakoni, V., Corneil, D., & Brea, J. (2018, July). Eligibility Traces and Plasticity on Behavioral Time Scales: Experimental Support of NeoHebbian Three-Factor Learning Rules. *Frontiers in Neural Circuits*, *12*. Retrieved 2019-08-09, from https://www.ncbi.nlm.nih.gov/pmc/

articles/PMC6079224/ doi: 10.3389/fncir.2018.00053

Gibbon, J. (1977). Scalar expectancy theory and Weber's law in animal timing. *Psychological Review*, *84*(3), 279–325. doi: 10.1037/0033-295X.84.3.279

Goel, A., & Buonomano, D. (2016, August). Temporal Interval Learning in Cortical Cultures Is Encoded in Intrinsic Network Dynamics. *Neuron*, *91*(2), 320–327. Retrieved 2016-08-12, from http://dx.doi.org/10.1016/j.neuron.2016.05.042 doi: 10.1016/j.neuron.2016.05.042

Goel, A., & Buonomano, D. V. (2014). Timing as an intrinsic property of neural networks: evidence from in vivo and in vitro experiments. *Philosophical transactions of the Royal Society B: Biological sciences*, *369*(1637), 20120460.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Goudar, V., & Buonomano, D. V. (2018, March). Encoding sensory and motor patterns as time-invariant trajectories in recurrent neural networks. *eLife*, *7*, e31134. Retrieved 2019-07-31, from https://doi.org/10.7554/eLife.31134 doi: 10.7554/eLife.31134

Goutagny, R., Jackson, J., & Williams, S. (2009, December). Self-generated theta oscillations in the hippocampus. *Nat Neurosci*, *12*(12), 1491–1493. Retrieved from http://dx.doi.org/10.1038/nn.2440 doi: 10.1038/nn.2440

Gouvêa, T. S., Monteiro, T., Motiwala, A., Soares, S., Machens, C., & Paton, J. J. (2015, December). Striatal dynamics explain duration judgments. *eLife*, *4*, e11386. Retrieved 2019-08-07, from https://doi.org/10.7554/eLife.11386 doi: 10.7554/eLife.11386

Grondin, S. (2010, April). Timing and time perception: A review of recent behavioral and neuroscience findings and theoretical

directions. *Attention, Perception, & Psychophysics*, *72*(3), 561–582. Retrieved 2019-05-21, from https://doi.org/10.3758/APP.72.3.561 doi: 10.3758/APP.72.3.561

Guerguiev, J., Lillicrap, T. P., & Richards, B. A. (2017, December). Towards deep learning with segregated dendrites. *eLife*, *6*, e22901. Retrieved 2019-08-03, from https://elifesciences.org/articles/22901 doi: 10.7554/eLife.22901

Guevara, M. R., Glass, L., Mackey, M. C., & Shrier, A. (1983, September). Chaos in neurobiology. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-13*(5), 790–798. doi: 10.1109/TSMC.1983.6313073

Gupta, M., Jin, L., & Homma, N. (2004). *Static and dynamic neural networks: from fundamentals to advanced theory*. John Wiley & Sons.

Hahnloser, R. H., Sarpeshkar, R., Mahowald, M. A., Douglas, R. J., & Seung, H. S. (2000). Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, *405*(6789), 947.

Han, M., Shi, Z., & Wang, W. (2004). Modeling Dynamic System by Recurrent Neural Network with State Variables. In F.-L. Yin, J. Wang, & C. Guo (Eds.), *Advances in Neural Networks - ISNN 2004* (pp. 200–205). Springer Berlin Heidelberg.

Hardy, N. F., Goudar, V., Romero-Sosa, J. L., & Buonomano, D. V. (2018, November). A model of temporal scaling correctly predicts that motor timing improves with speed. *Nature Communications*, *9*(1), 4732. Retrieved from https://doi.org/10.1038/s41467-018-07161-6 doi: 10.1038/s41467-018-07161-6

Harnad, S. (2005). To cognitze is to categorize: cognition is categorization. In *Handbook of Categorization in Cognitive Science* (pp. 19–43). Elsevier. Retrieved 2019-08-15, from https://linkinghub.elsevier.com/retrieve/pii/B9780080446127500561 doi: 10

.1016/B978-008044612-7/50056-1

Harrington, D. L., & Haaland, K. Y. (1999). Neural underpinnings of temporal processing: A review of focal lesion, pharmacological, and functional imaging research. *Reviews in the Neurosciences*, *10*(2), 91–116.

Harrington, D. L., Haaland, K. Y., & Hermanowitz, N. (1998). Temporal processing in the basal ganglia. *Neuropsychology*, *12*(1), 3.

Harvey, C. D., Coen, P., & Tank, D. W. (2012, April). Choice-specific sequences in parietal cortex during a virtual-navigation decision task. *Nature*, *484*(7392), 62–68. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3321074/ doi: 10.1038/nature10918

Harvey, C. D., Collman, F., Dombeck, D. A., & Tank, D. W. (2009). Intracellular dynamics of hippocampal place cells during virtual navigation. *Nature*, *461*(7266), 941–946.

Haykin, S. (2002). Adaptive filter theory. *Prentice Hall*, *2*, 478–481.

He, K., Huertas, M., Hong, S., Tie, X., Hell, J., Shouval, H., & Kirkwood, A. (2015, November). Distinct Eligibility Traces for LTP and LTD in Cortical Synapses. *Neuron*, *88*(3), 528–538. Retrieved 2019-08-09, from http://www.sciencedirect.com/science/article/pii/S0896627315008260 doi: 10.1016/j.neuron.2015.09.037

Hebb, D. O. (2005). *The organization of behavior: A neuropsychological theory*. Psychology Press.

Hendler, J. (2008, March). Avoiding Another AI Winter. *IEEE Intelligent Systems*, *23*(2), 2–4. doi: 10.1109/MIS.2008.20

Heys, J. G., & Dombeck, D. A. (2018, November). Evidence for a subcircuit in medial entorhinal cortex representing elapsed time during immobility. *Nature Neuroscience*, *21*(11), 1574. Re-

trieved 2019-06-06, from https://www.nature.com/articles/s41593-018-0252-8/ doi: 10.1038/s41593-018-0252-8

Hickok, G., & Poeppel, D. (2007, May). The cortical organization of speech processing. *Nature Reviews Neuroscience*, *8*(5), 393–402. Retrieved 2019-07-31, from http://www.nature.com/articles/nrn2113 doi: 10.1038/nrn2113

Hinton, S. C., & Meck, W. H. (2004, October). Frontal–striatal circuitry activated by human peak-interval timing in the supra-seconds range. *Cognitive Brain Research*, *21*(2), 171–182. Retrieved 2019-08-06, from http://www.sciencedirect.com/science/article/pii/S0926641004002198 doi: 10.1016/j.cogbrainres.2004.08.005

Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Hoerzer, G. M., Legenstein, R., & Maass, W. (2014). Emergence of complex computational structures from chaotic neural networks through reward-modulated Hebbian learning. *Cerebral cortex*, *24*(3), 677–690.

Hooper, S. L. (2001). Central Pattern Generators. In *eLS*. John Wiley & Sons, Ltd. Retrieved from http://dx.doi.org/10.1038/npg.els.0000032

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, *79*(8), 2554–2558.

Huertas, M. A., Schwettmann, S. E., & Shouval, H. Z. (2016). The Role of Multiple Neuromodulators in Reinforcement Learn-

ing That Is Based on Competition between Eligibility Traces. *Frontiers in Synaptic Neuroscience*, *8*. Retrieved 2019-08-09, from https://www.frontiersin.org/articles/10.3389/fnsyn.2016.00037/full doi: 10.3389/fnsyn.2016.00037

Hunsberger, E., & Eliasmith, C. (2016). Training Spiking Deep Networks for Neuromorphic Hardware. *arXiv:1611.05141 [cs]*. Retrieved 2019-08-03, from http://arxiv.org/abs/1611.05141 (arXiv: 1611.05141) doi: 10.13140/RG.2.2.10967.06566

Häusler, S., Markram, H., & Maass, W. (2003). Perspectives of the high-dimensional dynamics of neural microcircuits from the point of view of low-dimensional readouts. *Complexity*, *8*(4), 39–50.

Hutcheon, B., & Yarom, Y. (2000). Resonance, oscillation and the intrinsic frequency preferences of neurons. *Trends in neurosciences*, *23*(5), 216–222.

Ižikevič, E. M. (2010). *Dynamical systems in neuroscience: the geometry of excitability and bursting* (1. MIT Press paperback ed ed.). Cambridge, Mass.: MIT Press. (OCLC: 699607639)

Ingrosso, A., & Abbott, L. F. (2019, August). Training dynamically balanced excitatory-inhibitory networks. *PLOS ONE*, *14*(8), e0220547. Retrieved 2019-08-29, from https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0220547 doi: 10.1371/journal.pone.0220547

Ishii, K., van der Zant, T., Becanovic, V., & Ploger, P. (2004). Optimization of parameters of echo state network and its application to underwater robot. In (Vol. 3, pp. 2800–2805). IEEE.

Ivry, R. B., & Keele, S. W. (1989). Timing functions of the cerebellum. *Journal of Cognitive Neuroscience*, *1*(2), 136–152.

Ivry, R. B., & Schlerf, J. E. (2008, July). Dedicated and intrinsic mod-

els of time perception. *Trends in Cognitive Sciences*, *12*(7), 273–280. Retrieved from http://www.sciencedirect.com/science/article/pii/S1364661308001307 doi: 10.1016/j.tics.2008.04.002

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

Jaeger, H. (2002). Adaptive nonlinear system identification with echo state networks. In (pp. 593–600).

Jaeger, H., & Haas, H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *science*, *304*(5667), 78–80.

Jang, E. V. (2015). *Recurrent Neural Networks : a Dynamical Systems Perspective.*

Janssen, P., & Shadlen, M. N. (2005, February). A representation of the hazard rate of elapsed time in macaque area LIP. *Nature Neuroscience*, *8*(2), 234–241. Retrieved 2019-08-14, from https://www.nature.com/articles/nn1386 doi: 10.1038/nn1386

Jarne, C., & Laje, R. (2019, June). A detailed study of recurrent neural networks used to model tasks in the cerebral cortex. *arXiv:1906.01094 [cs, q-bio]*. Retrieved 2019-07-27, from http://arxiv.org/abs/1906.01094 (arXiv: 1906.01094)

Jarvis, S., Rotter, S., & Egert, U. (2010). Extending stability through hierarchical clusters in echo state networks. *Frontiers in neuroinformatics*, *4*.

Jeffress, L. A. (1948). A place theory of sound localization. *Journal of comparative and physiological psychology*, *41*(1), 35.

Jensen, O., & Colgin, L. L. (2007). Cross-frequency coupling between

neuronal oscillations. *Trends in Cognitive Sciences*, *11*(7), 267–269.

Jin, X., & Costa, R. M. (2015, August). Shaping action sequences in basal ganglia circuits. *Current Opinion in Neurobiology*, *33*, 188–196. Retrieved 2019-08-09, from http://www.sciencedirect.com/science/article/pii/S0959438815001051 doi: 10.1016/j.conb.2015.06.011

Johnston, A., Arnold, D. H., & Nishida, S. (2006). Spatially localized distortions of event time. *Current biology*, *16*(5), 472–479.

Joshi, P., & Maass, W. (2005). Movement generation with circuits of spiking neurons. *Neural Computation*, *17*(8), 1715–1738.

Kadmon, J., & Sompolinsky, H. (2015, November). Transition to Chaos in Random Neuronal Networks. *Physical Review X*, *5*(4), 041030. Retrieved 2019-08-05, from https://link.aps.org/doi/10.1103/PhysRevX.5.041030 doi: 10.1103/PhysRevX.5.041030

Kahana, M. J. (2006, August). The Cognitive Correlates of Human Brain Oscillations. *Journal of Neuroscience*, *26*(6), 1669–1672. Retrieved 2019-05-28, from http://www.jneurosci.org/content/26/6/1669 doi: 10.1523/JNEUROSCI.3737-05c.2006

Kalmbach, B. E., Davis, T., Ohyama, T., Riusech, F., Nores, W. L., & Mauk, M. D. (2010, February). Cerebellar Cortex Contributions to the Expression and Timing of Conditioned Eyelid Responses. *Journal of Neurophysiology*, *103*(4), 2039–2049. Retrieved 2019-08-09, from https://www.physiology.org/doi/full/10.1152/jn.00033.2010 doi: 10.1152/jn.00033.2010

Kandel, E. R., Schwartz, J. H., Jessell, T. M., Department of Biochemistry and Molecular Biophysics Thomas Jessell, Siegelbaum, S., & Hudspeth, A. (2000). *Principles of neural science* (Vol. 4). McGraw-hill New York.

Karmarkar, U. R., & Buonomano, D. V. (2007, Febru-

ary). Timing in the Absence of Clocks: Encoding Time in Neural Network States. *Neuron*, *53*(3), 427–438. Retrieved from http://www.sciencedirect.com/science/article/pii/S0896627307000256 doi: 10.1016/j.neuron.2007.01.006

Kastner, K. (2019, March). *Audio tools for numpy/python.* Retrieved from https://gist.github.com/kastnerkyle/179d6e9a88202ab0a2fe

Kastner, S., Pinsk, M. A., De Weerd, P., Desimone, R., & Ungerleider, L. G. (1999, April). Increased Activity in Human Visual Cortex during Directed Attention in the Absence of Visual Stimulation. *Neuron*, *22*(4), 751–761. Retrieved 2019-08-13, from http://www.sciencedirect.com/science/article/pii/S0896627300807345 doi: 10.1016/S0896-6273(00)80734-5

Katlowitz, K. A., Picardo, M. A., & Long, M. A. (2018). Stable Sequential Activity Underlying the Maintenance of a Precisely Executed Skilled Behavior. *Neuron*. Retrieved 2018-06-06, from http://dx.doi.org/10.1016/j.neuron.2018.05.017 doi: 10.1016/j.neuron.2018.05.017

Kawai, R., Markman, T., Poddar, R., Ko, R., Fantana, A., Dhawale, A., ... Ölveczky, B. (2015, May). Motor Cortex Is Required for Learning but Not for Executing a Motor Skill. *Neuron*, *86*(3), 800–812. Retrieved 2019-08-07, from http://www.sciencedirect.com/science/article/pii/S0896627315002202 doi: 10.1016/j.neuron.2015.03.024

Kayser, C., Montemurro, M. A., Logothetis, N. K., & Panzeri, S. (2009). Spike-phase coding boosts and stabilizes information carried by spatial and temporal spike patterns. *Neuron*, *61*(4), 597–608.

Keller, C. J., Honey, C. J., Mégevand, P., Entz, L., Ulbert, I., & Mehta,

A. D. (2014). Mapping human brain networks with cortico-cortical evoked potentials. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *369*(1653), 20130528.

Kempter, R., Gerstner, W., & van Hemmen, J. L. (1999, April). Hebbian learning and spiking neurons. *Physical Review E*, *59*(4), 4498–4514. Retrieved 2019-07-24, from https://link.aps.org/doi/10.1103/PhysRevE.59.4498 doi: 10.1103/PhysRevE.59.4498

Kietzmann, T. C., McClure, P., & Kriegeskorte, N. (2019, January). Deep Neural Networks in Computational Neuroscience. In *Oxford Research Encyclopedia of Neuroscience.* Oxford University Press. Retrieved 2019-08-03, from http://oxfordre.com/neuroscience/view/10.1093/acrefore/9780190264086.001.0001/acrefore-9780190264086-e-46 doi: 10.1093/acrefore/9780190264086.013.46

Kim, S. S., Rouault, H., Druckmann, S., & Jayaraman, V. (2017, May). Ring attractor dynamics in the Drosophila central brain. *Science*, *356*(6340), 849–853. Retrieved 2019-08-14, from http://www.sciencemag.org/lookup/doi/10.1126/science.aal4835 doi: 10.1126/science.aal4835

Klukas, M., Lewis, M., & Fiete, I. (2019, March). Flexible representation and memory of higher-dimensional cognitive variables with grid cells. *bioRxiv*. Retrieved 2019-05-28, from http://biorxiv.org/lookup/doi/10.1101/578641 doi: 10.1101/578641

Kobayashi, S., & Schultz, W. (2008, July). Influence of Reward Delays on Responses of Dopamine Neurons. *Journal of Neuroscience*, *28*(31), 7837–7846. Retrieved 2019-08-06, from https://www.jneurosci.org/content/28/31/7837 doi: 10.1523/JNEUROSCI.1600-08.2008

Kononowicz, T. W., & van Wassenhove, V. (2016). In Search of Os-

cillatory Traces of the Internal Clock. *Frontiers in Psychology*, *7*. Retrieved 2019-05-28, from https://www.frontiersin.org/articles/10.3389/fpsyg.2016.00224/full doi: 10.3389/fpsyg.2016.00224

Kosko, B. (1988). Bidirectional associative memories. *IEEE Transactions on Systems, man, and Cybernetics*, *18*(1), 49–60.

Kubota, K., & Niki, H. (1971, May). Prefrontal cortical unit activity and delayed alternation performance in monkeys. *Journal of Neurophysiology*, *34*(3), 337–347. Retrieved 2019-08-13, from https://www.physiology.org/doi/abs/10.1152/jn.1971.34.3.337 doi: 10.1152/jn.1971.34.3.337

Kuebler, E. S., Calderini, M., Longtin, A., Bent, N., Vincent-Lamarre, P., & Thivierge, J.-P. (2018, December). Non-monotonic accumulation of spike time variance during membrane potential oscillations. *Biological Cybernetics*, *112*(6), 539–545. Retrieved 2019-08-22, from https://doi.org/10.1007/s00422-018-0782-x doi: 10.1007/s00422-018-0782-x

Laje, R., & Buonomano, D. V. (2013, July). Robust timing and motor patterns by taming chaos in recurrent neural networks. *Nat Neurosci*, *16*(7), 925–933. Retrieved from http://dx.doi.org/10.1038/nn.3405

Lajoie, G., Lin, K. K., & Shea-Brown, E. (2013). Chaos and reliability in balanced spiking networks with temporal drive. *Physical Review E*, *87*(5), 052901.

Lajoie, G., Thivierge, J.-P., & Shea-Brown, E. (2014). Structured chaos shapes spike-response noise entropy in balanced neural networks. *Frontiers in Computational Neuroscience*, *8*. Retrieved from http://www.frontiersin.org/Journal/Abstract.aspx?s=237&name=computational_neuroscience&ART_DOI=10.3389/

fncom.2014.00123 doi: 10.3389/fncom.2014.00123

Lake, J. I., & Meck, W. H. (2013, January). Differential effects of amphetamine and haloperidol on temporal reproduction: Dopaminergic regulation of attention and clock speed. *Neuropsychologia*, *51*(2), 284–292. Retrieved 2019-08-06, from http://www.sciencedirect.com/science/article/pii/S0028393212003867 doi: 10.1016/j.neuropsychologia.2012.09.014

Lampl, I., & Yarom, Y. (1997, March). Subthreshold oscillations and resonant behavior: two manifestations of the same mechanism. *Neuroscience*, *78*(2), 325–341. Retrieved 2019-08-21, from http://www.sciencedirect.com/science/article/pii/S030645229600588X doi: 10.1016/S0306-4522(96)00588-X

Laurent, G., & Davidowitz, H. (1994, September). Encoding of olfactory information with oscillating neural assemblies. *Science (New York, N.Y.)*, *265*(5180), 1872–1875. doi: 10.1126/science.265.5180.1872

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436.

Lee, A. K., & Wilson, M. A. (2002, December). Memory of Sequential Experience in the Hippocampus during Slow Wave Sleep. *Neuron*, *36*(6), 1183–1194. Retrieved 2019-07-25, from http://www.sciencedirect.com/science/article/pii/S0896627302010966 doi: 10.1016/S0896-6273(02)01096-6

Legenstein, R., & Maass, W. (2007, April). Edge of chaos and prediction of computational performance for neural circuit models. *Echo State Networks and Liquid State Machines*, *20*(3), 323–334. Retrieved from http://www.sciencedirect.com/science/article/pii/S0893608007000433 doi: 10.1016/j.neunet.2007.04

.017

Lengyel, M., Szatmáry, Z., & Érdi, P. (2003). Dynamically detuned oscillations account for the coupled rate and temporal code of place cell firing. *Hippocampus*, *13*(6), 700–714. Retrieved 2019-07-25, from https://onlinelibrary.wiley.com/doi/abs/10.1002/hipo.10116 doi: 10.1002/hipo.10116

Leon, M. I., & Shadlen, M. N. (2003, April). Representation of Time by Neurons in the Posterior Parietal Cortex of the Macaque. *Neuron*, *38*(2), 317–327. Retrieved 2019-08-14, from http://www.sciencedirect.com/science/article/pii/S0896627303001855 doi: 10.1016/S0896-6273(03)00185-5

Leung, L.-W. S., & Yim, C.-Y. C. (1991, July). Intrinsic membrane potential oscillations in hippocampal neurons in vitro. *Brain Research*, *553*(2), 261–274. Retrieved 2019-08-21, from http://www.sciencedirect.com/science/article/pii/000689939190834I doi: 10.1016/0006-8993(91)90834-I

Levitin, D. J. (2009). The neural correlates of temporal structure in music. *Music and medicine*, *1*(1), 9–13.

Lewis, P. A., & Miall, R. (2006). A right hemispheric prefrontal system for cognitive time measurement. *Behavioural Processes*, *71*(2-3), 226–234.

Li, X., Zhong, L., Xue, F., & Zhang, A. (2015). A Priori Data-Driven Multi-Clustered Reservoir Generation Algorithm for Echo State Network. *PloS one*, *10*(4).

Licklider, J. C. R. (1951, January). A Duplex Theory of Pitch Perception. *The Journal of the Acoustical Society of America*, *23*(1), 147–147. Retrieved 2019-08-01, from https://asa.scitation.org/doi/abs/10.1121/1.1917296 doi: 10.1121/1.1917296

Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016,

December). Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*, *7*(1), 13276. Retrieved 2019-08-03, from http://www.nature.com/articles/ncomms13276 doi: 10.1038/ncomms13276

Lin, X., Yang, Z., & Song, Y. (2008). The application of echo state network in stock data mining. In (pp. 932–937). Springer.

Lisman, J., & Buzsáki, G. (2008). A neural coding scheme formed by the combined function of gamma and theta oscillations. *Schizophrenia bulletin*, *34*(5), 974–980.

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

Long, M. A., & Fee, M. S. (2008, November). Using temperature to analyse temporal dynamics in the songbird motor pathway. *Nature*, *456*(7219), 189–194. Retrieved 2019-08-01, from http://www.nature.com/articles/nature07448 doi: 10.1038/nature07448

Long, M. A., Jin, D. Z., & Fee, M. S. (2010, November). Support for a synaptic chain model of neuronal sequence generation. *Nature*, *468*(7322), 394–399. Retrieved 2019-08-01, from https://www.nature.com/articles/nature09514 doi: 10.1038/nature09514

Lüthi, A., & McCormick, D. A. (1998, March). Periodicity of Thalamic Synchronized Oscillations: the Role of Ca2+-Mediated Upregulation of Ih. *Neuron*, *20*(3), 553–563. Retrieved 2019-08-21, from http://www.sciencedirect.com/science/article/pii/S0896627300809940 doi: 10.1016/S0896-6273(00)80994-0

Lukoševičius, M., & Jaeger, H. (2009, August). Reservoir computing approaches to recurrent neural network train-

ing. *Computer Science Review*, *3*(3), 127–149. Retrieved from http://www.sciencedirect.com/science/article/pii/S1574013709000173 doi: 10.1016/j.cosrev.2009.03.005

Lustig, C., Matell, M. S., & Meck, W. H. (2003). Not "just" a coincidence: Frontal-striatal interactions in working memory and interval timing. *Memory*, *13*(3-4), 441–448.

Maass, W., Joshi, P., & Sontag, E. D. (2007, January). Computational Aspects of Feedback in Neural Circuits. *PLoS Comput Biol*, *3*(1), e165. Retrieved from http://dx.plos.org/10.1371%2Fjournal.pcbi.0020165 doi: 10.1371/journal.pcbi.0020165

Maass, W., Natschläger, T., & Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, *14*(11), 2531–2560.

Maass, W., Natschläger, T., & Markram, H. (2004). Computational models for generic cortical microcircuits. *Computational neuroscience: A comprehensive approach*, *18*, 575.

Macar, F., Coull, J., & Vidal, F. (2006). The supplementary motor area in motor and perceptual time processing: fMRI studies. *Cognitive processing*, *7*(2), 89–94.

MacKay, W. (2004, December). Wheels of Motion: Oscillatory Potentials in the Motor Cortex. In *Motor Cortex in Voluntary Movements* (Vol. 20044553). CRC Press. Retrieved 2019-07-28, from http://www.crcnetbase.com/doi/abs/10.1201/9780203503584.ch7 doi: 10.1201/9780203503584.ch7

Mainen, Z. F., & Sejnowski, T. J. (1995). Reliability of spike timing in neocortical neurons. *Science*, *268*(5216), 1503–1506.

Mante, V., Sussillo, D., Shenoy, K. V., & Newsome, W. T. (2013, November). Context-dependent computation by recurrent dy-

namics in prefrontal cortex. *Nature*, *503*(7474), 78–84. Retrieved from http://dx.doi.org/10.1038/nature12742

Marder, E., & Bucher, D. (2001, November). Central pattern generators and the control of rhythmic movements. *Current Biology*, *11*(23), R986–R996. Retrieved 2019-07-28, from http://www.sciencedirect.com/science/article/pii/S0960982201005814 doi: 10.1016/S0960-9822(01)00581-4

Marder, E., & Calabrese, R. L. (1996, July). Principles of rhythmic motor pattern generation. *Physiological Reviews*, *76*(3), 687–717. Retrieved 2019-07-28, from https://www.physiology.org/doi/abs/10.1152/physrev.1996.76.3.687 doi: 10.1152/physrev.1996.76.3.687

Maricq, A. V., & Church, R. M. (1983). The differential effects of haloperidol and methamphetamine on time estimation in the rat. *Psychopharmacology*, *79*(1), 10–15. doi: 10.1007/bf00433008

Markram, H., Lübke, J., Frotscher, M., & Sakmann, B. (1997, January). Regulation of Synaptic Efficacy by Coincidence of Postsynaptic APs and EPSPs. *Science*, *275*(5297), 213–215. Retrieved 2019-08-09, from https://science.sciencemag.org/content/275/5297/213 doi: 10.1126/science.275.5297.213

Matell, M. S. (2014). Searching for the Holy Grail: Temporally Informative Firing Patterns in the Rat. In *Neurobiology of Interval Timing* (pp. 209–234). Springer.

Matell, M. S., & Meck, W. H. (2000). Neuropsychological mechanisms of interval timing behavior. *BioEssays*, *22*(1), 94–103. Retrieved 2019-05-21, from https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291521-1878%28200001%2922%3A1%3C94%3A%3AAID-BIES14%3E3.0.CO%3B2-E doi: 10.1002/(SICI)1521-1878(200001)22:1<94::AID-BIES14>3.0.CO;2-E

Matell, M. S., & Meck, W. H. (2004). Cortico-striatal circuits and interval timing: coincidence detection of oscillatory processes. *Cognitive Brain Research*, *21*(2), 139–170.

Mauk, M. D., & Buonomano, D. V. (2004). The neural basis of temporal processing. *Annu. Rev. Neurosci.*, *27*, 307–340.

Mazurek, M. E., Roitman, J. D., Ditterich, J., & Shadlen, M. N. (2003). A role for neural integrators in perceptual decision making. *Cerebral cortex*, *13*(11), 1257–1269.

McCulloch, W. S., & Pitts, W. (1943, December). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, *5*(4), 115–133. Retrieved 2019-08-02, from https://doi.org/10.1007/BF02478259 doi: 10.1007/BF02478259

Meck, W. H., & Malapani, C. (2004, October). Neuroimaging of interval timing. *Cognitive Brain Research*, *21*(2), 133–137. Retrieved 2019-08-06, from http://www.sciencedirect.com/science/article/pii/S0926641004002186 doi: 10.1016/j.cogbrainres.2004.07.010

Meck, W. H., Penney, T. B., & Pouthas, V. (2008, April). Cortico-striatal representation of time in animals and humans. *Current Opinion in Neurobiology*, *18*(2), 145–152. Retrieved 2019-08-07, from http://www.sciencedirect.com/science/article/pii/S0959438808000755 doi: 10.1016/j.conb.2008.08.002

Medina, J. F., & Mauk, M. D. (2000, November). Computer simulation of cerebellar information processing. *Nature Neuroscience*, *3*(S11), 1205–1211. Retrieved 2019-08-09, from http://www.nature.com/articles/nn1100_1205 doi: 10.1038/81486

Mello, G. B., Soares, S., & Paton, J. J. (2015). A scalable population code for time in the striatum. *Current Biology*, *25*(9), 1113–1122.

Meshulam, L., Gauthier, J. L., Brody, C. D., Tank, D. W., & Bialek,

W. (2017, December). Collective Behavior of Place and Non-place Neurons in the Hippocampal Network. *Neuron*, *96*(5), 1178–1191.e4. Retrieved 2019-08-23, from https://linkinghub.elsevier.com/retrieve/pii/S0896627317309960 doi: 10.1016/j.neuron.2017.10.027

Meunier, D., Lambiotte, R., & Bullmore, E. T. (2010). Modular and hierarchically modular organization of brain networks. *Frontiers in neuroscience*, *4*.

Miall, C. (1989). The storage of time intervals using oscillating neurons. *Neural Computation*, *1*(3), 359–371.

Milnor, J. (1985). On the concept of attractor. In *The theory of chaotic attractors* (pp. 243–264). Springer.

Minsky, M. (1961). Steps toward artificial intelligence. *Proceedings of the IRE*, *49*(1), 8–30.

Minsky, M., & Papert, S. (1969). An introduction to computational geometry. *Cambridge tiass., HIT*.

Molgedey, L., Schuchhardt, J., & Schuster, H. G. (1992, December). Suppressing chaos in neural networks by noise. *Physical Review Letters*, *69*(26), 3717–3719. Retrieved 2019-05-24, from https://link.aps.org/doi/10.1103/PhysRevLett.69.3717 doi: 10.1103/PhysRevLett.69.3717

Montgomery, S. M., Betancur, M. I., & Buzsáki, G. (2009, February). Behavior-Dependent Coordination of Multiple Theta Dipoles in the Hippocampus. *Journal of Neuroscience*, *29*(5), 1381–1394. Retrieved 2019-07-26, from https://www.jneurosci.org/content/29/5/1381 doi: 10.1523/JNEUROSCI.4339-08.2009

Morrone, M. C., Ross, J., & Burr, D. (2005). Saccadic eye movements cause compression of time as well as space. *Nature neuroscience*, *8*(7), 950.

Moser, E. I., Kropff, E., & Moser, M.-B. (2008, July). Place Cells, Grid Cells, and the Brain's Spatial Representation System. *Annual Review of Neuroscience*, *31*(1), 69–89. Retrieved 2019-06-27, from http://www.annualreviews.org/doi/10.1146/annurev.neuro.31.061307.090723 doi: 10.1146/annurev.neuro.31.061307.090723

Mountcastle, V. B. (1997). The columnar organization of the neocortex. *Brain*, *120*(4), 701–722.

Naud, R., Houtman, D., Rose, G. J., & Longtin, A. (2015, September). Counting on dis-inhibition: a circuit motif for interval counting and selectivity in the anuran auditory system. *Journal of Neurophysiology*, *114*(5), 2804–2815. Retrieved 2019-12-11, from https://www.physiology.org/doi/full/10.1152/jn.00138.2015 doi: 10.1152/jn.00138.2015

Nicola, W., & Clopath, C. (2017, December). Supervised learning in spiking neural networks with FORCE training. *Nature Communications*, *8*(1), 2208. Retrieved 2019-07-21, from https://www.nature.com/articles/s41467-017-01827-3 doi: 10.1038/s41467-017-01827-3

Nicola, W., & Clopath, C. (2019, July). A diversity of interneurons and Hebbian plasticity facilitate rapid compressible learning in the hippocampus. *Nature Neuroscience*, *22*(7), 1168. Retrieved 2019-07-21, from https://www.nature.com/articles/s41593-019-0415-2 doi: 10.1038/s41593-019-0415-2

Ohyama, T., Nores, W. L., Murphy, M., & Mauk, M. D. (2003, April). What the cerebellum computes. *Trends in Neurosciences*, *26*(4), 222–227. Retrieved 2019-08-09, from http://www.sciencedirect.com/science/article/pii/S0166223603000547 doi: 10.1016/S0166-2236(03)00054-7

O'Keefe, J. (1976, January). Place units in the hippocampus of the freely moving rat. *Experimental Neurology*, *51*(1), 78–109. Retrieved 2019-06-09, from https://linkinghub.elsevier.com/retrieve/pii/0014488676900558 doi: 10.1016/0014-4886(76)90055-8

O'Keefe, J., & Recce, M. L. (1993). Phase relationship between hippocampal place units and the EEG theta rhythm. *Hippocampus*, *3*(3), 317–330.

Pakkenberg, B., Pelvig, D., Marner, L., Bundgaard, M. J., Gundersen, H. J. G., Nyengaard, J. R., & Regeur, L. (2003, January). Aging and the human neocortex. *Proceedings of the 6th International Symposium on the Neurobiology and Neuroendocrinology of Aging*, *38*(1–2), 95–99. Retrieved from http://www.sciencedirect.com/science/article/pii/S0531556502001511 doi: 10.1016/S0531-5565(02)00151-1

Pandarinath, C., O'Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., ... Sussillo, D. (2018, October). Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, *15*(10), 805–815. Retrieved 2019-07-22, from http://www.nature.com/articles/s41592-018-0109-9 doi: 10.1038/s41592-018-0109-9

Pascanu, R., Mikolov, T., & Bengio, Y. (2012). Understanding the exploding gradient problem. *CoRR, abs/1211.5063*, *2*.

Pastalkova, E., Itskov, V., Amarasingham, A., & Buzsáki, G. (2008, September). Internally Generated Cell Assembly Sequences in the Rat Hippocampus. *Science*, *321*(5894), 1322–1327. Retrieved 2019-07-21, from https://science.sciencemag.org/content/321/5894/1322 doi: 10.1126/science.1159775

Paton, J. J., & Buonomano, D. V. (2018). The Neural Basis of Timing:

Distributed Mechanisms for Diverse Functions. *Neuron*, *98*(4), 687–705.

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

Perrett, S. P., Ruiz, B. P., & Mauk, M. D. (1993). Cerebellar cortex lesions disrupt learning-dependent timing of conditioned eyelid responses. *The Journal of Neuroscience*, *13*(4), 1708–1718.

Peters, A. J., Chen, S. X., & Komiyama, T. (2014, June). Emergence of reproducible spatiotemporal activity during motor learning. *Nature*, *510*(7504), 263–267. Retrieved 2019-08-01, from https://www.nature.com/articles/nature13235 doi: 10.1038/nature13235

Petter, E. A., Lusk, N. A., Hesslow, G., & Meck, W. H. (2016). Interactive roles of the cerebellum and striatum in sub-second and supra-second timing: support for an initiation, continuation, adjustment, and termination (ICAT) model of temporal processing. *Neuroscience & Biobehavioral Reviews*, *71*, 739–755.

Rabinovich, M., Huerta, R., & Laurent, G. (2008, July). Transient Dynamics for Neural Processing. *Science*, *321*(5885), 48. Retrieved from http://science.sciencemag.org/content/321/5885/48.abstract doi: 10.1126/science.1155564

Rajan, K., Abbott, L., & Sompolinsky, H. (2010). Stimulus-dependent suppression of chaos in recurrent neural networks. *Physical Review E*, *82*(1), 011903.

Rajan, K., Harvey, C. D., & Tank, D. W. (2016). Recurrent network models of sequence generation and memory. *Neuron*.

Rakitin, B. C., Gibbon, J., Penney, T. B., Malapani, C., Hinton, S. C., & Meck, W. H. (1998). Scalar expectancy theory and peak-interval

timing in humans. *Journal of Experimental Psychology: Animal Behavior Processes*, *24*(1), 15.

Rao, S. M., Mayer, A. R., & Harrington, D. L. (2001, March). The evolution of brain activation during temporal processing. *Nature Neuroscience*, *4*(3), 317–323. Retrieved 2019-08-06, from http://www.nature.com/articles/nn0301_317 doi: 10.1038/85191

Raymond, J. L., & Medina, J. F. (2018). Computational Principles of Supervised Learning in the Cerebellum. *Annual Review of Neuroscience*, *41*(1), 233–253. Retrieved 2019-08-09, from https://doi.org/10.1146/annurev-neuro-080317-061948 doi: 10.1146/annurev-neuro-080317-061948

Reppert, S. M., & Weaver, D. R. (2002). Coordination of circadian timing in mammals. *Nature*, *418*(6901), 935.

Richards, B. A., & Lillicrap, T. P. (2019). Dendritic solutions to the credit assignment problem. *Current opinion in neurobiology*, *54*, 28–36.

Rodieck, R. W. (1965, December). Quantitative analysis of cat retinal ganglion cell response to visual stimuli. *Vision Research*, *5*(12), 583–601. Retrieved 2019-07-31, from http://www.sciencedirect.com/science/article/pii/0042698965900337 doi: 10.1016/0042-6989(65)90033-7

Roitman, J. D., & Shadlen, M. N. (2002, November). Response of Neurons in the Lateral Intraparietal Area during a Combined Visual Discrimination Reaction Time Task. *Journal of Neuroscience*, *22*(21), 9475–9489. Retrieved 2019-08-14, from https://www.jneurosci.org/content/22/21/9475 doi: 10.1523/JNEUROSCI.22-21-09475.2002

Rokni, U., & Sompolinsky, H. (2011, October). How the Brain Generates Movement. *Neural Computation*, *24*(2), 289–331. Retrieved

2019-07-27, from https://doi.org/10.1162/NECO_a_00223 doi: 10.1162/NECO_a_00223

Romo, R., Brody, C. D., Hernandez, A., & Lemus, L. (1999, June). Neuronal correlates of parametric working memory in the prefrontal cortex. *Nature*, *399*(6735), 470–473. Retrieved from http://dx.doi.org/10.1038/20939 doi: 10.1038/20939

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

Rubinov, M., & Sporns, O. (2010). Complex network measures of brain connectivity: uses and interpretations. *Neuroimage*, *52*(3), 1059–1069.

Rubinov, M., Sporns, O., Thivierge, J.-P., & Breakspear, M. (2011). Neurobiologically realistic determinants of self-organized criticality in networks of spiking neurons. *PLoS computational biology*, *7*(6), e1002038.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive modeling*, *5*(3), 1.

Runyan, C. A., Piasini, E., Panzeri, S., & Harvey, C. D. (2017, July). Distinct timescales of population coding across cortex. *Nature*, *advance online publication*. Retrieved from http://dx.doi.org/10.1038/nature23020

Sainath, T. N., Vinyals, O., Senior, A., & Sak, H. (2015, April). Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4580–4584). doi: 10.1109/ICASSP.2015.7178838

Sainburg, T. (2018, July). *Spectrograms, MFCCs, and Inversion in*

Python. Retrieved from https://timsainburg.com/python-mel-compression-inversion.html

Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2017, December). Recent Advances in Recurrent Neural Networks. *arXiv:1801.01078 [cs]*. Retrieved 2019-07-27, from http://arxiv.org/abs/1801.01078 (arXiv: 1801.01078)

Schaefer, A. T., Angelo, K., Spors, H., & Margrie, T. W. (2006). Neuronal oscillations enhance stimulus discrimination by ensuring action potential precision.

Schmidhuber, J. (2015, January). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117. Retrieved 2019-08-02, from http://www.sciencedirect.com/science/article/pii/S0893608014002135 doi: 10.1016/j.neunet.2014.09.003

Schmidt, R., Diba, K., Leibold, C., Schmitz, D., Buzsáki, G., & Kempter, R. (2009, October). Single-Trial Phase Precession in the Hippocampus. *Journal of Neuroscience*, *29*(42), 13232–13241. Retrieved 2019-07-24, from https://www.jneurosci.org/content/29/42/13232 doi: 10.1523/JNEUROSCI.2270-09.2009

Schmidt, R. A., Lee, T. D., Winstein, C., Wulf, G., & Zelaznik, H. N. (2018). *Motor control and learning: A behavioral emphasis*. Human kinetics.

Schmitz, D., Gloveli, T., Behr, J., Dugladze, T., & Heinemann, U. (1998, August). Subthreshold membrane potential oscillations in neurons of deep layers of the entorhinal cortex. *Neuroscience*, *85*(4), 999–1004. Retrieved 2019-08-22, from http://www.sciencedirect.com/science/article/pii/S0306452298001134 doi: 10.1016/S0306-4522(98)00113-4

Schrauwen, B., Verstraeten, D., & Van Campenhout, J. (2007). An overview of reservoir computing: theory, applications and im-

plementations. In (pp. 471–482).

Schultz, W. (1998). Predictive reward signal of dopamine neurons. *Journal of neurophysiology*, *80*(1), 1–27.

Schultz, W. (2016). Dopamine reward prediction error coding. *Dialogues in Clinical Neuroscience*, *18*(1), 23.

Seidemann, E., Meilijson, I., Abeles, M., Bergman, H., & Vaadia, E. (1996, January). Simultaneously recorded single units in the frontal cortex go through sequences of discrete and stable states in monkeys performing a delayed localization task. *Journal of Neuroscience*, *16*(2), 752–768. Retrieved 2019-08-14, from https://www.jneurosci.org/content/16/2/752 doi: 10.1523/JNEUROSCI.16-02-00752.1996

Serences, J. T., Ester, E. F., Vogel, E. K., & Awh, E. (2009, February). Stimulus-Specific Delay Activity in Human Primary Visual Cortex. *Psychological Science*, *20*(2), 207–214. Retrieved 2019-08-13, from https://doi.org/10.1111/j.1467-9280.2009.02276.x doi: 10.1111/j.1467-9280.2009.02276.x

Sheng, M.-j., Lu, D., Shen, Z.-m., & Poo, M.-m. (2019, May). Emergence of stable striatal D1r and D2r neuronal ensembles with distinct firing sequence during motor learning. *Proceedings of the National Academy of Sciences*, *116*(22), 11038–11047. Retrieved 2019-05-29, from https://www.pnas.org/content/116/22/11038 doi: 10.1073/pnas.1901712116

Shuler, M. G., & Bear, M. F. (2006, March). Reward Timing in the Primary Visual Cortex. *Science*, *311*(5767), 1606. Retrieved from http://science.sciencemag.org/content/311/5767/1606.abstract doi: 10.1126/science.1123513

Simen, P., Rivest, F., Ludvig, E. A., Balci, F., & Killeen, P. (2013). Timescale Invariance in the Pacemaker-Accumulator

Family of Timing Models. *Timing & Time Perception*, *1*(2), 159–188. Retrieved 2019-08-01, from https://brill.com/view/journals/time/1/2/article-p159_3.xml doi: 10.1163/22134468-00002018

Skowronski, M. D., & Harris, J. G. (2007). Automatic speech recognition using a predictive echo state network classifier. *Neural networks*, *20*(3), 414–423.

Soares, S., Atallah, B. V., & Paton, J. J. (2016, December). Midbrain dopamine neurons control judgment of time. *Science*, *354*(6317), 1273–1277. Retrieved 2019-08-06, from https://science.sciencemag.org/content/354/6317/1273 doi: 10.1126/science.aah5234

Soderstrand, M. A., Jenkins, W. K., Jullien, G. A., & Taylor, F. J. (Eds.). (1986). *Residue Number System Arithmetic: Modern Applications in Digital Signal Processing*. Piscataway, NJ, USA: IEEE Press.

Soltesz, I., & Deschenes, M. (1993). Low-and high-frequency membrane potential oscillations during theta activity in CA1 and CA3 pyramidal neurons of the rat hippocampus under ketamine-xylazine anesthesia. *Journal of neurophysiology*, *70*(1), 97–116.

Sompolinsky, H., Crisanti, A., & Sommers, H. (1988). Chaos in random neural networks. *Physical Review Letters*, *61*(3), 259.

Song, H. F., Yang, G. R., & Wang, X.-J. (2016, February). Training Excitatory-Inhibitory Recurrent Neural Networks for Cognitive Tasks: A Simple and Flexible Framework. *PLOS Computational Biology*, *12*(2), e1004792. Retrieved 2019-07-22, from https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004792 doi: 10.1371/journal.pcbi.1004792

Song, S. C., Beatty, J. A., & Wilson, C. J. (2016). The ionic mechanism of membrane potential oscillations and membrane resonance in striatal LTS interneurons. *Journal of Neurophysiology*, *116*(4), 1752–1764.

Sussillo, D. (2014, April). Neural circuits as computational dynamical systems. *Theoretical and computational neuroscience*, *25*(0), 156–163. Retrieved from http://www.sciencedirect.com/science/article/pii/S0959438814000166 doi: 10.1016/j.conb.2014.01.008

Sussillo, D., & Abbott, L. (2009, August). Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, *63*(4), 544–557. Retrieved from http://www.sciencedirect.com/science/article/pii/S0896627309005479 doi: 10.1016/j.neuron.2009.07.018

Sussillo, D., & Abbott, L. (2012, May). Transferring Learning from External to Internal Weights in Echo-State Networks with Sparse Connectivity. *PLoS ONE*, *7*(5), e37372. Retrieved from http://dx.doi.org/10.1371%2Fjournal.pone.0037372 doi: 10.1371/journal.pone.0037372

Sussillo, D., & Barak, O. (2013). Opening the black box: low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural computation*, *25*(3), 626–649.

Sussillo, D., Churchland, M. M., Kaufman, M. T., & Shenoy, K. V. (2015). A neural network that finds a naturalistic solution for the production of muscle activity. *Nature neuroscience*, *18*(7), 1025–1033.

Tanaka, G., Yamane, T., Héroux, J. B., Nakane, R., Kanazawa, N., Takeda, S., ... Hirose, A. (2018). Recent Advances in Physical Reservoir Computing: A Review. *arXiv preprint*

arXiv:1808.04962.

Taube, J. S., & Muller, R. U. (1998). Comparisons of head direction cell activity in the postsubiculum and anterior thalamus of freely moving rats. *Hippocampus*, *8*(2), 87–108. Retrieved 2019-08-14, from https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291098-1063%281998%298%3A2%3C87%3A%3AAID-HIPO1%3E3.0.CO%3B2-4 doi: 10.1002/(SICI)1098-1063(1998)8:2<87::AID-HIPO1>3.0.CO;2-4

Tavanaei, A., & Maida, A. S. (2016, November). Bio-Inspired Spiking Convolutional Neural Network using Layer-wise Sparse Coding and STDP Learning. *arXiv:1611.03000 [cs]*. Retrieved 2019-08-03, from http://arxiv.org/abs/1611.03000 (arXiv: 1611.03000)

Tchernichovski, O., Mitra, P. P., Lints, T., & Nottebohm, F. (2001). Dynamics of the vocal imitation process: how a zebra finch learns its song. *Science*, *291*(5513), 2564–2569.

Teki, S., Grube, M., Kumar, S., & Griffiths, T. D. (2011, March). Distinct Neural Substrates of Duration-Based and Beat-Based Auditory Timing. *Journal of Neuroscience*, *31*(10), 3805–3812. Retrieved 2019-08-09, from https://www.jneurosci.org/content/31/10/3805 doi: 10.1523/JNEUROSCI.5561-10.2011

Thivierge, J.-P., Comas, R., & Longtin, A. (2014). Attractor dynamics in local neuronal networks. *Frontiers in Neural Circuits*, *8*. Retrieved 2019-06-04, from https://www.frontiersin.org/articles/10.3389/fncir.2014.00022/full doi: 10.3389/fncir.2014.00022

Tiesinga, P., Fellous, J.-M., & Sejnowski, T. J. (2008). Regulation of spike timing in visual cortical circuits. *Nature reviews neuroscience*, *9*(2), 97–107.

Toledo-Suárez, C., Duarte, R., & Morrison, A. (2014). Liquid com-

puting on and off the edge of chaos with a striatal microcircuit. *Frontiers in computational neuroscience*, *8*.

Tort, A. B. L., Kramer, M. A., Thorn, C., Gibson, D. J., Kubota, Y., Graybiel, A. M., & Kopell, N. J. (2008, December). Dynamic cross-frequency couplings of local field potential oscillations in rat striatum and hippocampus during performance of a T-maze task. *Proceedings of the National Academy of Sciences*, *105*(51), 20517–20522. Retrieved 2019-08-22, from https://www.pnas.org/content/105/51/20517 doi: 10.1073/pnas.0810524105

Traub, R. D., Miles, R., & Wong, R. K. S. (1989). Model of the origin of rhythmic population oscillations in the hippocampal slice. *Science*, *243*(4896), 1319–1325.

Treisman, M. (1963). Temporal discrimination and the indifference interval: Implications for a model of the "internal clock". *Psychological Monographs: General and Applied*, *77*(13), 1–31. Retrieved 2019-08-01, from http://doi.apa.org/getdoi.cfm?doi=10.1037/h0093864 doi: 10.1037/h0093864

Trengove, C., van Leeuwen, C., & Diesmann, M. (2013, April). High-capacity embedding of synfire chains in a cortical network model. *Journal of Computational Neuroscience*, *34*(2), 185–209. Retrieved 2019-08-01, from https://doi.org/10.1007/s10827-012-0413-9 doi: 10.1007/s10827-012-0413-9

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

van Vreeswijk, C. v., & Sompolinsky, H. (1998). Chaotic balanced state in a model of cortical circuits. *Neural computation*, *10*(6), 1321–1371.

Verstraeten, D., Schrauwen, B., & Stroobandt, D. (2006, July).

Reservoir-based techniques for speech recognition. In *The 2006 IEEE International Joint Conference on Neural Network Proceedings* (pp. 1050–1053). doi: 10.1109/IJCNN.2006.246804

Verzi, S. J., Rothganger, F., Parekh, O. D., Quach, T.-T., Miner, N. E., Vineyard, C. M., ... Aimone, J. B. (2018, October). Computing with Spikes: The Advantage of Fine-Grained Timing. *Neural Computation*, *30*(10), 2660–2690. Retrieved 2019-08-24, from https://www.mitpressjournals.org/doi/abs/10.1162/neco_a_01113 doi: 10.1162/neco_a_01113

Vincent, K., Tauskela, J., & Thivierge, J.-P. (2012). Extracting functionally feedforward networks from a population of spiking neurons. *Frontiers in Computational Neuroscience*, *6*. Retrieved from http://www.frontiersin.org/Journal/Abstract.aspx?s=237&name=computational_neuroscience&ART_DOI=10.3389/fncom.2012.00086 doi: 10.3389/fncom.2012.00086

Vincent-Lamarre, P., Lajoie, G., & Thivierge, J.-P. (2016). Driving reservoir models with oscillations: a solution to the extreme structural sensitivity of chaotic networks. *Journal of computational neuroscience*, *41*(3), 305–322.

Waddington, C. H. (2014). *The strategy of the genes*. Routledge.

Wang, X.-J. (2010). Neurophysiological and computational principles of cortical rhythms in cognition. *Physiological reviews*, *90*(3), 1195–1268.

Wang, X.-J. (2016, July). Synaptic reverberation underlying mnemonic persistent activity. *Trends in Neurosciences*, *24*(8), 455–463. Retrieved 2016-07-09, from http://dx.doi.org/10.1016/S0166-2236(00)01868-3 doi: 10.1016/S0166-2236(00)01868-3

Weissenberger, F., Meier, F., Lengler, J., Einarsson, H., & Steger, A. (2017, September). Long Synfire Chains Emerge by Spike-

Timing Dependent Plasticity Modulated by Population Activity. *International Journal of Neural Systems*, *27*(08), 1750044. Retrieved 2019-08-01, from https://www.worldscientific.com/doi/abs/10.1142/S0129065717500447 doi: 10.1142/S0129065717500447

Westlye, L. T., Walhovd, K. B., Dale, A. M., Bjørnerud, A., Due-Tønnessen, P., Engvig, A., ... Fjell, A. M. (2009). Life-span changes of the human brain white matter: diffusion tensor imaging (DTI) and volumetry. *Cerebral cortex*, bhp280.

Widrow, B., & Hoff, M. E. (1962). Associative storage and retrieval of digital information in networks of adaptive "neurons". In *Biological Prototypes and Synthetic Systems* (pp. 160–160). Springer.

Wilson, H. R., & Cowan, J. D. (1973, September). A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik*, *13*(2), 55–80. Retrieved 2019-08-14, from https://doi.org/10.1007/BF00288786 doi: 10.1007/BF00288786

Wimmer, K., Nykamp, D. Q., Constantinidis, C., & Compte, A. (2014, March). Bump attractor dynamics in prefrontal cortex explains behavioral precision in spatial working memory. *Nature Neuroscience*, *17*(3), 431–439. Retrieved 2019-08-14, from https://www.nature.com/articles/nn.3645 doi: 10.1038/nn.3645

Wittmann, M., Leland, D. S., & Paulus, M. P. (2007, June). Time and decision making: differential contribution of the posterior insular cortex and the striatum during a delay discounting task. *Experimental Brain Research*, *179*(4), 643–653. Retrieved 2019-08-06, from https://doi.org/10.1007/s00221-006-0822-y doi: 10.1007/s00221-006-0822-y

Yamazaki, T., & Tanaka, S. (2007). The cerebellum as a liquid state machine. *Neural Networks*, *20*(3), 290–297.

Yamins, D. L. K., & DiCarlo, J. J. (2016, March). Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, *19*(3), 356–365. Retrieved 2019-07-31, from https://www.nature.com/articles/nn.4244 doi: 10.1038/nn.4244

Yuste, R. (2015, August). From the neuron doctrine to neural networks. *Nature Reviews Neuroscience*, *16*(8), 487–497. Retrieved 2019-08-02, from https://www.nature.com/articles/nrn3962 doi: 10.1038/nrn3962

Yuste, R., MacLean, J. N., Smith, J., & Lansner, A. (2005). The cortex as a central pattern generator. *Nature Reviews Neuroscience*, *6*(6), 477–483.

Zhiping, W. Y. J. N. X. (2009). effects of spectral radius on echo-state-network's training. Internet Computing for Science and Engineering (ICICSE), 2009 Fourth International Conference on.

Zugaro, M. B., Arleo, A., Berthoz, A., & Wiener, S. I. (2003, April). Rapid Spatial Reorientation and Head Direction Cells. *Journal of Neuroscience*, *23*(8), 3478–3482. Retrieved 2019-08-14, from https://www.jneurosci.org/content/23/8/3478 doi: 10.1523/JNEUROSCI.23-08-03478.2003