

Spatiotemporal Computations of an Excitable and Plastic Brain: Neuronal Plasticity Leads to Noise-Robust and Noise-Constructive Computations

Hazem Toutounji*, Gordon Pipa

Institute of Cognitive Science, University of Osnabrück, Osnabrück, Lower Saxony, Germany

Abstract

It is a long-established fact that neuronal plasticity occupies the central role in generating neural function and computation. Nevertheless, no unifying account exists of how neurons in a recurrent cortical network learn to compute on temporally and spatially extended stimuli. However, these stimuli constitute the norm, rather than the exception, of the brain's input. Here, we introduce a geometric theory of learning spatiotemporal computations through neuronal plasticity. To that end, we rigorously formulate the problem of neural representations as a relation in space between stimulus-induced neural activity and the asymptotic dynamics of excitable cortical networks. Backed up by computer simulations and numerical analysis, we show that two canonical and widely spread forms of neuronal plasticity, that is, spike-timing-dependent synaptic plasticity and intrinsic plasticity, are both necessary for creating neural representations, such that these computations become realizable. Interestingly, the effects of these forms of plasticity on the emerging neural code relate to properties necessary for both combating and utilizing noise. The neural dynamics also exhibits features of the most likely stimulus in the network's spontaneous activity. These properties of the spatiotemporal neural code resulting from plasticity, having their grounding in nature, further consolidate the biological relevance of our findings.

Citation: Toutounji H, Pipa G (2014) Spatiotemporal Computations of an Excitable and Plastic Brain: Neuronal Plasticity Leads to Noise-Robust and Noise-Constructive Computations. *PLoS Comput Biol* 10(3): e1003512. doi:10.1371/journal.pcbi.1003512

Editor: Olaf Sporns, Indiana University, United States of America

Received: August 30, 2013; **Accepted:** January 29, 2014; **Published:** March 20, 2014

Copyright: © 2014 Toutounji, Pipa. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Funding: The authors acknowledge the financial support of the State Lower Saxony, Germany via the University of Osnabrück, and the European project PHOCUS in the Framework 'Information and Communication Technologies' (FP7-ICT-2009-C/Proposal Nr. 240763). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: htoutounji@uni-osnabrueck.de

Introduction

Neuronal plasticity, both homeostatic and synaptic, is the central ingredient for the generation and adaptation of neural function and computation [1]. However, it remains mostly unclear how neurons in recurrent neural networks utilize neuronal plasticity to self-organize and to learn computing on temporally and spatially extended stimuli [2–4].

A full grasp of the principles of self-organization by plasticity in recurrent neural networks is *jointly* hampered by the diversity of existing neuronal plasticity mechanisms [5–7] and the limited understanding of their functions and cooperations, by the *emergent* nature of computation in recurrent systems, in the sense that computation is a collective phenomenon of the system as a whole and cannot be fully understood from the contribution of individual neurons [8,9], and by the fact that neural systems are subject to noise [10–13]. In this paper, we simultaneously address these issues by studying the basic principles of self-organization in *recurrent* networks that arise from the interaction of synaptic and homeostatic intrinsic *plasticity*, and given that the network is subject to *noise*. To this end, we use numerical methods to explore the dynamics of nonautonomous, i.e. stimulus-driven, and plastic recurrent networks, and we provide a mathematical formalization for attaining a rigorously sound perspective (see Methods).

Incorporating synaptic plasticity with homeostasis goes back to Bienenstock, Cooper, and Monro's groundbreaking work known

as the *BCM theory* [14]. Through rigorous mathematical analysis, the BCM theory predicted the necessity of a certain form of a sliding threshold, i.e. a homeostatic adjustment of neuronal excitability, for stabilizing the plastic afferent weights of a single neuron. Empirical findings supported the hypothesis of adjustable excitability and showed that it manifests through changes of neuronal properties at the soma [6,7]. While the BCM theory suggests homeostasis as a stabilization mechanism of synaptic weights with no direct influence on the neuron's encoding properties, Triesch proposed a homeostatic *intrinsic plasticity* (IP) mechanism that increases the neuron's encoding capacity and cooperates with *synaptic plasticity* (SP) to discover nonlinear independent features of the neuron's inputs [15].

These investigations, among others [16,17], are very insightful in pinpointing how synaptic and homeostatic plasticity interact in *single neurons*. In addition, *feedforward* neural networks greatly simplify the analysis and understanding of self-organization and computation based on neuronal plasticity. For such architectures, both single plasticity rules, as well as combinations of different plasticity mechanisms, had been linked to neural computation, such as the formation of receptive fields [14], the related identification of statistically-independent components [15,17,18], and predictive coding [19]. However, it is important to note that neurons are embedded within large and highly recurrent networks [20–23], and that an efficient use of neuronal resources entails distributed encoding schemes [8,9]. In addition, besides the spatial

Author Summary

The world is not perceived as a chain of segmented sensory still lifes. Instead, it appears that the brain is capable of integrating the temporal dependencies of the incoming sensory stream with the spatial aspects of that input. It then transfers the resulting whole in a useful manner, in order to reach a coherent and causally sound image of our physical surroundings, and to act within it. These spatiotemporal computations are made possible through a cluster of local and coexisting adaptation mechanisms known collectively as neuronal plasticity. While this role is widely known and supported by experimental evidence, no unifying theory of how the brain, through the interaction of plasticity mechanisms, gets to represent spatiotemporal computations in its spatiotemporal activity. In this paper, we aim at such a theory. We develop a rigorous mathematical formalism of spatiotemporal representations within the input-driven dynamics of cortical networks. We demonstrate that the interaction of two of the most common plasticity mechanisms, intrinsic and synaptic plasticity, leads to representations that allow for spatiotemporal computations. We also show that these representations are structured to tolerate noise and to even benefit from it.

features of the world, its temporal structure should also be captured by the neural code [4,24–26].

Our understanding of neural information processing would greatly improve by extending the principles of self-organization to recurrent neural circuits, since the latter constitute the basic computational units in the cortex [22]. Lazar et al. were the first to study the emergence of computation from the interaction of different forms of plasticity on recurrent neural networks [27,28]. This study builds on their findings. However, we *do not* restrict the definition of computation to linear classifiers of the *reservoir computing* (RC) paradigm [3,29,30]. In addition to training linear classifiers for measuring the *computational performance*, we identified the necessity of analyzing the response of the recurrent neural network itself as an *input-driven dynamical system* [31,32], and of concurrently viewing the network as a *communication channel* by taking an *information-theoretical* perspective [33]. Combining these tools enables us to understand how information is encoded in recurrent systems, how such encoding is developing from self-organization, and how noise is effecting both.

Analyzing the dynamics of a *large* and, most importantly, *input-driven* neural system shaped by biologically-relevant *plasticity* is a hard task due to several methodological constraints. First, most analysis tools from *dynamical systems theory* are confined to small dynamical systems with very few degrees of freedom [34]. Exceptions are studies that circumvent this limitation by focusing on the low-dimensional collective dynamics of neural networks, e.g., [35], or studies that probe the high-dimensional phase space of the neural network, such as the classic example of Hopfield Networks [36]. Other instances of high-dimensional dynamical systems include ring networks and their coexisting periodic attractors [37], stable heteroclinic orbits [38,39], unstable periodic attractors [40], and others [41–43].

The second and most important methodological constraint is that the use of standard dynamical systems theory is inappropriate, since it deals with autonomous systems only, i.e. systems with no explicit dependence on time. In reality, however, neural networks are subject to a flux of ever changing stimulation that renders them nonautonomous. A theory of *nonautonomous dynamical systems* is only

recently taking shape as a branch of applied mathematics [31,32]. The fields of neural computation and computational biology are constantly contributing to the theory with concepts such as meta-transients and attractor morphing [37,44], γ -systems [45], and the nonautonomous dynamics of echo state networks [46].

A simple intuition of the difference between nonautonomous and autonomous systems can be stated as follows. Attractors of an autonomous dynamical system are defined by the system alone, and are therefore fixed. In contrast, attractors of a nonautonomous system are jointly defined by the dynamical system and its input. As the input changes, so does the attractor landscape of the system. This highlights the fact that studying computations in a driven system using the methods of autonomous dynamical systems is insufficient, since the input-induced changes of the system, i.e. changes of its attractor landscape, are ignored in that case.

The third constraint is that the complexity of the dynamics increases due to the neural system's adaptability. The presence of plasticity imposes restrictions on the dynamics a network can exhibit, thus keeping the network dynamics in a regime that can support complex computations. To the best of our knowledge, no attempt prior to this work has been taken to combine high-dimensionality and nonautonomy with the consequences of plasticity on dynamics. We demonstrate that plasticity *sculpts* the stimulus-specific dynamic landscapes, and by that, serves in improving representation of the provided input. Moreover, neuronal plasticity can adapt and learn stimulus-induced sequences of such stimulus-specific landscapes. We thereby show that neuronal plasticity improves spatiotemporal computations.

Given the above, we highlight and explain that spatiotemporal computations require two basic ingredients: a homeostatic mechanism that regulates neuronal activity, and synaptic learning that adapts the network's recurrent connectivity to the stimulus. We show that combining both types leads to a system that: first, learns the temporal structure of the input and carries out nonlinear computations, second, is noise tolerant, and third, even benefits from the presence of noise that sets the system to an input-sensitive dynamic regime.

The paper is structured as follows. We first characterize the effects of self-organized adaptation that is based on synaptic and homeostatic intrinsic plasticity and their combination. For that, we use tasks where both random and temporally-structured inputs are reconstructed and predicted, as well as a task where nonlinear computations are performed. We estimate the network's self-information capacity (its *entropy*), and its input-information capacity (the *mutual information* between the input and the network). We then interlude to qualitatively analyze the resulting dynamics of plastic changes based on the theory of nonautonomous dynamical systems. We explain the superior computation of conjoining synaptic and intrinsic plasticity based on both the informational and dynamical analyses. Building upon that, we study network noise, and demonstrate how noise is combated and exploited through the interaction of synaptic and intrinsic plasticity.

Results

In this section, we guide the reader through the following topics. We start by elucidating the computational power gained through the combination of synaptic and homeostatic plasticity mechanisms on recurrent neural networks of the *k-Winner-Take-All* (k WTA) type. We investigate the role of these plasticity forms in shaping the neural code through their effects on the informational and dynamical landscapes of the network. We conclude by illustrating how synaptically and homeostatically organized recurrent networks both benefit from noise and tolerate its presence. Figure 1

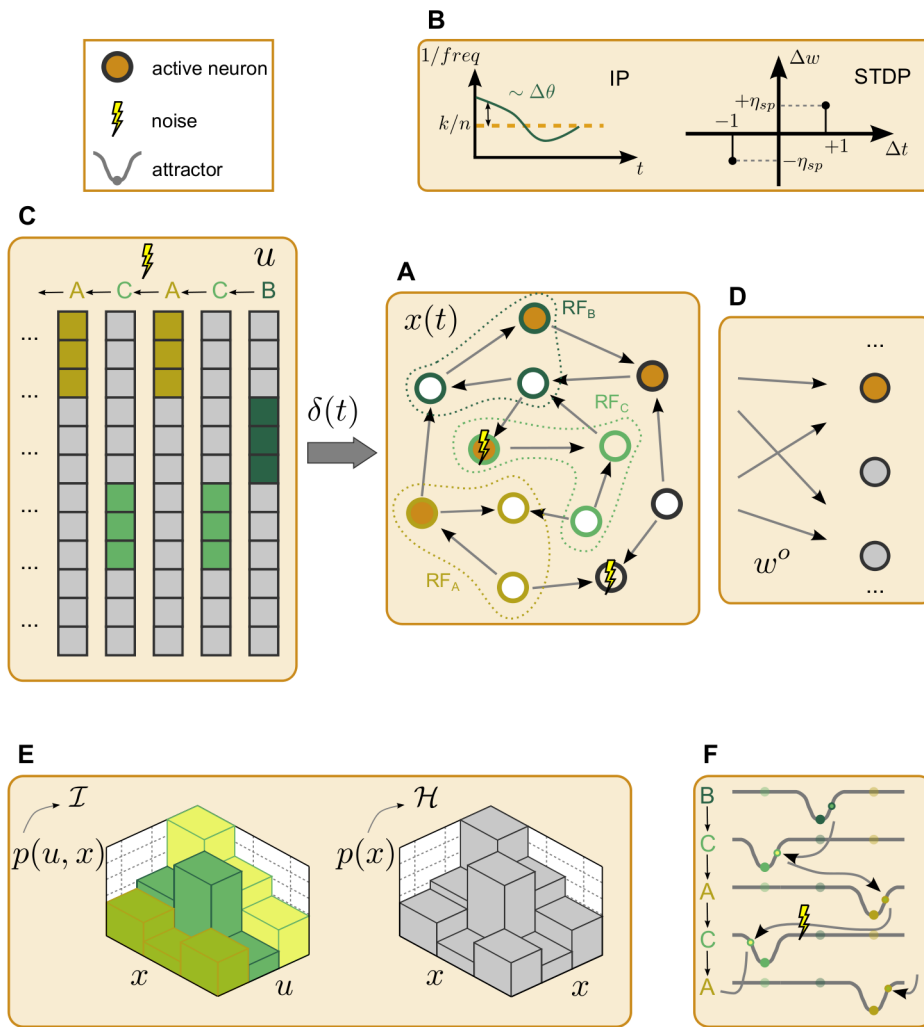


Figure 1. Overview of the recurrent network model and the methods for analyzing its computational capabilities. (A) An exemplary recurrent neural network of 12 neurons. The network state x has a 4-Winner-Take-All (4WTA) nonlinear dynamics, where the 4 neurons with the highest membrane potential fire and the rest are silent. The membrane potential is the sum of the recurrent afferents and the external drive δ . It is also depolarized (hyperpolarized) with decreasing (increasing) excitability threshold θ . The recurrent network can also be subject to *noise*, while reserving the 4WTA dynamics: when a neuron fails to spike due to noise, another fires instead. (B) The recurrent network is adapted by two plasticity mechanisms. The excitability threshold θ is modulated by *intrinsic plasticity* (IP), the recurrent afferents w by *spike-timing-dependent synaptic plasticity* (STDP). (C) The external drive δ consists of discrete symbols that follow a certain stochastic dynamics, and each projects to a corresponding *receptive field* (RF). The exemplary drive is a 3-symbols Markov chain $A \rightarrow B \rightarrow C$ that allows a probability for *noisy* transitions, i.e. $A \rightarrow C$. (D) Linear functions of the network state x parametrized by output weights w^o fitted to (possibly nonlinear) target functions of sequences of the external drive. (E) Nonlinear information-theoretic quantities are measured: network state entropy \mathcal{H} and the mutual information \mathcal{I} of the network state x and input sequence u . (F) Analysis of the appearance and disappearance of attractors due to the external drive within the network as an input-driven dynamical system.

doi:10.1371/journal.pcbi.1003512.g001

schematically illustrates the network model, the plasticity rules, and the formal probes we used to evaluate and describe the resulting computational properties. More details are available in the Methods section.

Computational Power

The interaction of different forms of plasticity produces a rather complex emergent behavior that cannot be explained trivially by the individual operation of each. We therefore start with exploring the effects induced by the combination of *spike-timing-dependent synaptic plasticity* (STDP) and *intrinsic plasticity* (IP). We compare the computational performance of recurrent networks trained either with both synaptic and intrinsic plasticity (SIP-RNs), with synaptic plasticity alone (SP-RNs), or with intrinsic plasticity alone (IP-RNs),

in addition to nonplastic recurrent networks, where the synaptic efficacies and firing thresholds are random.

Following the *plasticity phase*, a network is *reset* to random initial conditions and the *training phase* starts. Output weights from the recurrent network to linear readouts are computed with linear regression so that the readouts activity is the optimal linear classifier of a target signal. The target signal depends on the computational task. That is followed by the *testing phase*, at which performance is computed. Performance is measured by the percentage of correctly matched readout activity to the target signal.

Naturally, during simulation, the recurrent network is excited by a task-dependent external drive. The battery of tasks we deployed was designed to *abstract* a certain aspect of the spatiotemporal computations faced by biological brains, i.e.

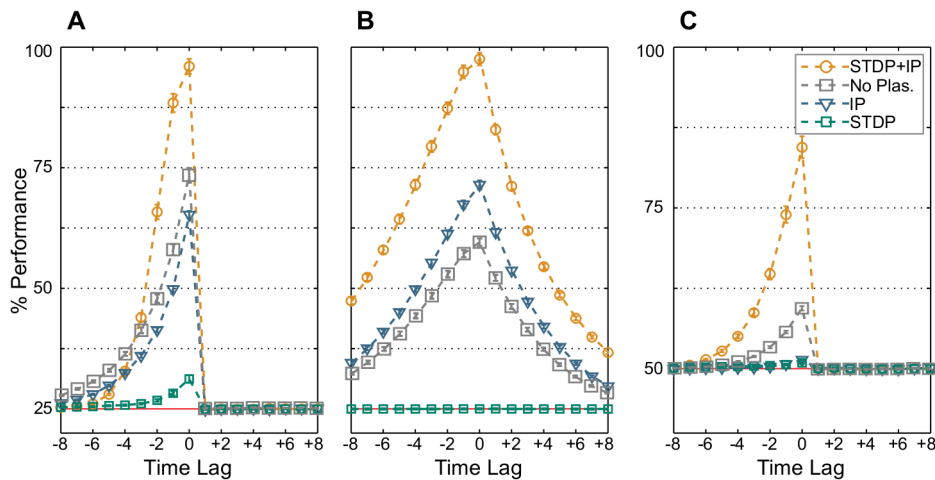


Figure 2. Average classification performance. 100 networks are trained by STDP and IP simultaneously (orange), IP alone (blue), STDP alone (green), or are nonplastic (gray). Optimal linear classifiers are then trained to perform (A) the memory task $\text{RAND} \times 4$, (B) the prediction task Markov-85 , and (C) the nonlinear task Parity-3 . Nonplastic networks have their weights trained by STDP and then randomly shuffled, so that they have the same weight and threshold distributions as SP-RNs. However, due to the shuffling, their weight matrices carry no structure. Error bars indicate standard error of the mean. The red line marks chance level. The x-axis shows the input time-lag. Negative time-lags indicate the past, and positive ones, the future.

doi:10.1371/journal.pcbi.1003512.g002

recalling past stimuli, predicting future ones, and nonlinearly transforming them. The memory task $\text{RAND} \times 4$, the prediction task Markov-85 , and the nonlinear task Parity-3 , as well as the plasticity models and simulation conditions, are detailed in the Methods section.

Figure 2 shows that SIP-RNs significantly outperform both IP-RNs and SP-RNs in all tasks. Inputs from 3 time steps in the past are successfully retained far beyond chance level in the memory task $\text{RAND} \times 4$ (Figure 2A). Understandably, performance drops to chance level for future stimuli (positive time-lags), since input symbols are equiprobable and their temporal succession carries no structure. Such is the case for the nonlinear task (Figure 2C). It is worth noting that solving the nonlinear task Parity-3 requires recalling three successive stimuli, which adds to the computational load. The recurrent network, through learning the temporally-structured input of the task Markov-85 , boosts the readouts' ability to reconstruct past symbols in comparison to the structureless memory task $\text{RAND} \times 4$. It also allows for the prediction of future stimuli far beyond chance (Figure 2B).

STDP alone fails to provide the recurrent network with means to encode necessary information. This leads to SP-RNs performing at almost chance level in all tasks. Intrinsic plasticity, on the other hand, endues recurrent networks with an intermediate ability to sustain past inputs (Figure 2A). IP-RNs also seem to learn the temporal structure of the input, as optimal linear classifiers are capable of predicting future stimuli (Figure 2B). Intrinsic plasticity is, however, insufficient for nonlinear computations, as IP-RNs barely perform above chance in the nonlinear parity task.

We also compare the performance of nonplastic kWTA networks with similar weight and threshold distributions as SP-RNs (shown in gray in Figure 2). They perform better than IP-RNs on the memory and nonlinear tasks, and worse on the prediction task. In all tasks, these nonplastic networks perform worse than SIP-RNs. We also show in Text S1 that nonplastic networks with comparable weight and threshold distributions as SIP-RNs also perform significantly lower than plastic networks. These results supply the evidence that the presence of plasticity enhances the computational power of recurrent neural networks (see Text S1 for

a discussion on heuristics for finding comparable random networks). No further analysis is carried out on these nonplastic networks, since the aim of this paper is to discern the effects of synaptic and intrinsic plasticity on spatiotemporal computations.

Neural Code

Explaining the superiority of networks modified by deploying both STDP and IP starts from isolating the individual role of each plasticity mechanism in defining the spatiotemporal neural code. In that regard, a well-informed intuition is that STDP learns the basic structure of the input as the connectivity resulting from STDP reflects the input sequence transitions. IP, on the other hand, increases the neural bandwidth by introducing redundancy to the code, as IP leads to the longest periodic cycles in the spontaneous activity of kWTA networks (See Figure 8 and Figure 4A in [27]).

The spatiotemporal neural code, or the *neural code* for short, can be characterized by both the *absolute capacity* of the network activity to store information and by how network activity *encodes* the spatially and temporally extended network input. *Entropy* of the network activity measures its absolute capacity, i.e. the repertoire of network states that the network can actually visit and potentially assign to some input sequence. The assignment of a network state to an input sequence means that this particular network state *encodes* or *represents* that input sequence. *Mutual information* between network input sequences and network states quantifies the extent of how successful this assignment is. Not every visited network state needs to be assigned an input sequence. A redundant code is reflected by input sequences being represented by multiple network states. Also, a network state might fail to encode an input, thus reflecting uninformative noise states.

We investigate the neural code characteristics of kWTA networks by estimating both the entropy of the network state and the mutual information between network input sequences and network states. We drive the network by $\text{RAND} \times 4$ input, and for computational tractability, we limit the estimation of mutual information to three-step inputs. An optimal encoder of this input sequence will then be a network with 6 bits of mutual information. The information-theoretical quantities are computed at intervals of the plasticity

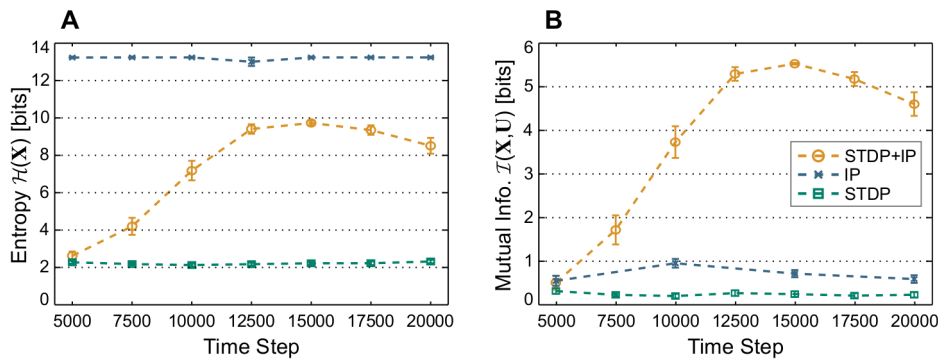


Figure 3. Network state entropy and the mutual information with input. (A) Network state entropy $\mathcal{H}(X)$ and (B) the mutual information with the three most recent $\text{RAND} \times 4$ inputs $\mathcal{I}(U, X)$ as they develop through the plasticity phase for SP-RNs (green), IP-RNs (blue), and SIP-RNs (orange). Mutual information for IP-RNs is estimated from 500000 time steps, and is averaged over 5 networks only. Other values are averaged over 50 networks and estimated from 100000 samples for each network. Error bars indicate standard error of the mean. doi:10.1371/journal.pcbi.1003512.g003

phase under the three plasticity conditions. At these intervals, the plastic variables are fixed and the driven network is reinitialized and run for a sufficient number of steps, and passed along with the input to the entropy and mutual information estimators. More details on how these measurements are carried out are found in the Methods section.

Figure 3 shows how these measures develop through the plasticity phase (For a discussion on the effects of longer plasticity exposure, see Text S2). SP-RNs' entropy remains constant at 2 bits. This means that SP-RNs visit only 4 network states (green in Figure 3A). However, these network states encode no information of the input sequence, as mutual information remains practically zero (green in Figure 3B). We call this 2 bits input-insensitive code the *minimal code*, as it captures no more than a single possible succession of the 4 inputs. This effect is the result of the interaction between the machination of STDP and the initial firing thresholds and weights configuration. Transitions, such as $A \rightarrow C$ in the input space, are to be stored in some of the synapses that connect neurons in the receptive field of $A(RF_A)$ with those in the receptive field of $C(RF_C)$. At each time step, one transition, such as $A \rightarrow C$, could be easier to reinforce with the causal (potentiating) side of STDP for RF_C neurons having little higher excitability (internal drive plus their own firing threshold). Without IP to tune down this excitability and with further contribution from the recurrency of the network, a positive feedback loop is generated, and this transition becomes more and more potentiated at the expense of others. This transition then becomes independent of the actual drive the network is receiving: the network becomes input-insensitive.

On the other side of the entropy spectrum, we find IP-RNs. Through IP's constant adjustment of the neuronal excitability, many neurons contribute to the neural code and IP-RNs visit a large number of states. Entropy and the network state bandwidth are the highest (blue in Figure 3A). One may view IP's effect as an introduction of *intrinsic deterministic noise* to the network activity. The increase in bandwidth of the network activity raises the odds for the random weights of an IP-RN to store an input sequence. In fact, many network states encode the same input sequence, resulting in a redundant code. However, without a synaptic reinforcement of representations, many states are visited due to the internal dynamics of the network, and not due to the external drive. These states remain uninformative and input sequences not successfully encoded: the mutual information (blue in Figure 3B), and hence the classification performance, are low.

The development of the neural code for SIP-RNs follows, however, a more interesting path. At the beginning, STDP has the upper hand and a 2 bits minimal code is generated. Through providing intrinsic deterministic noise, IP enriches the neural code by increasing redundancy and entropy (orange in Figure 3A). At the same time, STDP incrementally associates different network states to different input sequences by adjusting the synaptic weights as seen from the increase of mutual information (orange in Figure 3B). Then together, synaptic and homeostatic plasticity cooperate to create a code that is both *redundant* and *input-specific*. These properties are crucial for noise-robustness, as will be shown later in this text.

Post-Plasticity Perturbation

A dynamical system's behavior depends on its past activity. Therefore, testing a system requires assuming plausible initial conditions. The recurrent neural network at hand, even though it is small in comparison to a real neural circuit, has a number of possible initial conditions too large for all its initial conditions to be tested. So far, we have chosen random initial conditions for the network activity following the plasticity phase. From now on, we choose the initial conditions systematically by reinitializing the network activity depending on a *perturbation* π . This perturbation is applied to the end state of the plasticity phase, such that the end state of the plasticity phase and the initial state of the training phase are at a distance 2π from one another. For details of how the initial conditions are selected depending on the parameter π , we refer the reader to the Methods section.

To discern the effect of this perturbation, we compute the performance of the trained system with the three combinations of synaptic and intrinsic plasticity. We do this both for a system that is perturbed and for a system that starts from the last state that the dynamics reaches at the end of the preceding plasticity phase. We find no difference between the two cases of initial conditions for either IP-RNs or SP-RNs. However, when the neural network is trained by both synaptic and intrinsic plasticity (SIP-RNs), we find that the perturbed networks have better performance, as is illustrated in Figure 4A–C. The high performance of SIP-RNs that results from random initial conditions, as is shown in Figure 2, is easily explainable. It stems from the fact that random initialization is merely a large perturbation, since the probability of choosing a random state from such a large set of possibilities that is at a small distance from a particular region of the state space is insignificant, compared to a state that is at a large distance. Moreover, we find

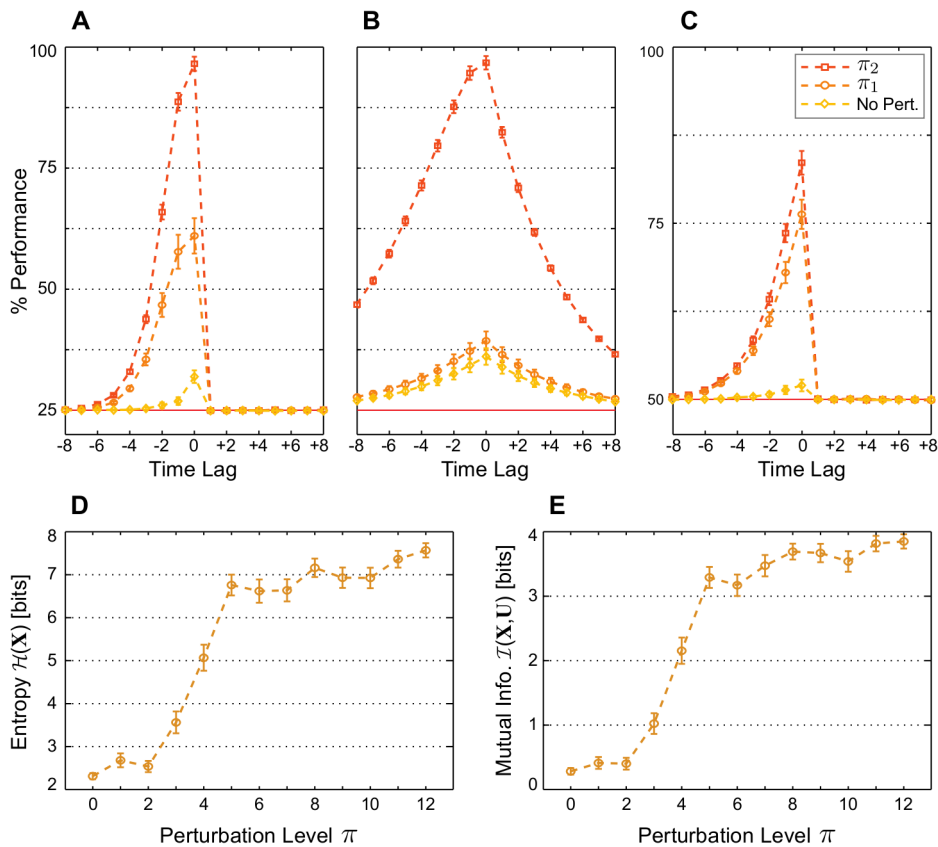


Figure 4. Post-plasticity perturbation. 100 networks are trained by STDP and IP simultaneously on (A) the memory task $\text{RAND} \times 4$, (B) the prediction task Markov-85 , and (C) the nonlinear task Parity-3 with increasing perturbation level: $\pi=0$ (yellow), $\pi=4$ (orange), and $\pi=12$ (red). Error bars indicate standard error of the mean. The red line marks chance level. The x-axis shows the input time-lag. Negative time-lags indicate the past, and positive ones, the future. (D) Network state entropy $\mathcal{H}(X)$ and (E) the mutual information with the three most recent $\text{RAND} \times 4$ inputs $\mathcal{I}(X, U)$ at the end of the plasticity phase for different perturbation levels. Values are averaged over 50 networks and estimated from 5000 samples for each network. Error bars indicate standard error of the mean. doi:10.1371/journal.pcbi.1003512.g004

that regardless of the task, larger perturbations result in higher average performance. This is also reflected in the neural code, where network state entropy and the mutual information with input correlate with higher perturbation (see Figure 4D–E).

This suggests that within the phase space of SIP-RNs there exist at least two dynamic regimes. Post-plasticity perturbation also provides the first sign of how SIP-RNs can benefit from noise, as it might put the system in the regime more suitable for computation.

Dynamic Regimes

Optimal linear classifiers show that kWTA networks equipped with both homeostatic and synaptic plasticity are capable of creating spatiotemporal codes and performing nonlinear computation. Measuring entropy and mutual information allows for a quantification of the emerging neural code. But what are the geometric features of the neural code that allow for such computations? How do network states *represent* the spatiotemporal input in a useful way? A major part of the Methods section is devoted to developing the mathematical formalization of discrete-time nonautonomous dynamical systems. References to definitions, a proposition, and a theorem from that section are featured in the following results, as we apply these concepts to our model neural network. We view this treatment not merely as an exercise in mathematics. It allows for a rigorous description of the computational properties emerging from plasticity that are

beyond the scrutiny of quantitative measures, such as linear classification performance and carried information. A consequence of these properties is also the two noise-related features we examine later.

For a formal treatment of spatiotemporal computations which result from plasticity, we need to extend the theory of nonautonomous dynamical systems to provide a notion for representations, to specify how these representations allow for computations, and to discern the effect of plasticity in enhancing these representations for the sake of computation. But first, we start by identifying the modes of operation, i.e. the dynamic regimes, the model plastic neural network has, since not all regimes might be suitable for computation.

According to Proposition 3 and Definition 6, when subject to stimulation, kWTA networks are *input-driven discrete-time dynamical systems*. For such systems, two extremes exist regarding the degree of sensitivity the system exhibits in response to its input. At one extreme, the system shows no change of response for different inputs, so that it follows its own dynamics, as if no input exists. In such a mode of operation, the system is *input-insensitive*. The other extreme is when the system's response is different for each input and initial condition. A single system can show, in principle, multiple modes of operation, depending on the initial conditions. The set of initial conditions that show a single mode of operation defines a dynamic regime and a basin of attraction.

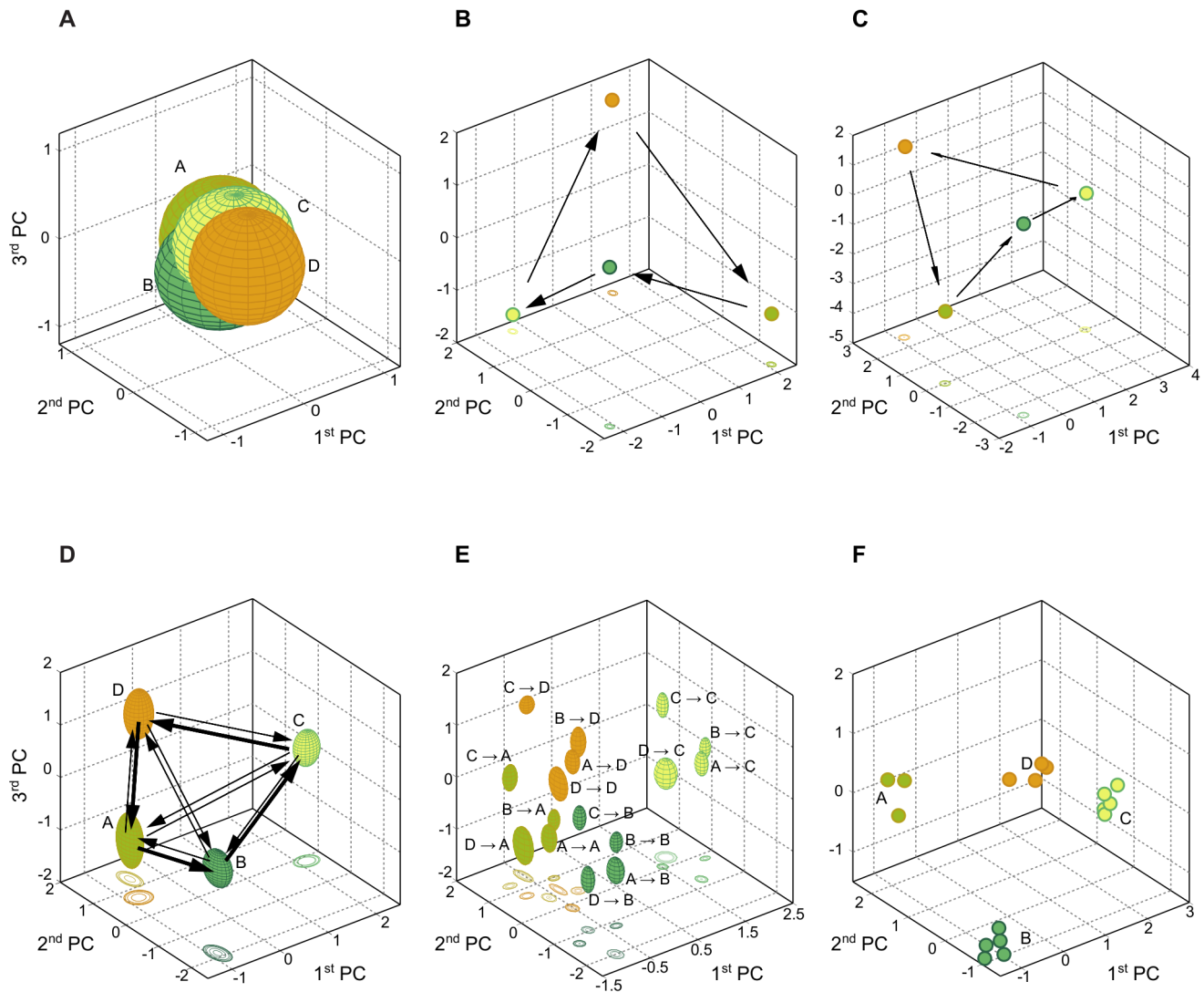


Figure 5. Plasticity effects on networks dynamics and input representations under the prediction task input. The three dimensions correspond to the first three principal components (PCs) of the network activity. (A) Highly-overlapping order-1 volumes of representation of an IP-RN. (B) Input-insensitive global attractor of a SP-RN that corresponds to a minimal code. (C) With no perturbed ($\pi=0$), a SIP-RN dynamics also converges to an input-insensitive attractor and exhibits a minimal code. (D) Approximate visualization of order-1 volumes of representation of a SIP-RN. The approximation uses the means and the standard deviations of the corresponding coordinates of the network activity in the principal components space as the center and semi-axes lengths of ellipsoids. Arrows correspond to the transitions from one input symbol to other. Their thickness symbolizes the probability of a transition, which reflects the Markov-85 transition probability. The collection of volumes of representation and the arrows show the perturbation set within which the nonautonomous attractor resides. (E) Order-2 volumes of representation of a SIP-RN also approximated using the mean and standard deviations of coordinates. Order-2 volumes are more exact approximations to the order-1 representations according to the volumes' inclusion property. The correspondence is clarified by using similar color coding. (F) Autonomous periodic attractors of a SIP-RN, each belonging to one of the autonomous semi-dynamical systems associated with one Markov-85 input. For clarity, no arrows are drawn between the vertexes of an attractor.
doi:10.1371/journal.pcbi.1003512.g005

In a first step, we visualize the high-dimensional response of the system to its input. To that end, we down-project the network activity to the first three principal components, and we study the effects of STDP and IP on the network's dynamics and input representations in this reduced 3-dimensional space (Figure 5). This analysis is performed on networks with Markov-85 input which fully demonstrate the relevant properties. It is important to note that while our analysis concerns the dynamics following the plasticity phase, we are still able to infer how it unfolds during this phase from the development of the neural code (Figure 3), as we make clear later.

As suggested by the performance of SP-RNs (Figure 2) and their neural code entropy and mutual information (Figure 3), their state space is dominated by an *input-insensitive* basin of attraction and these networks behave like autonomous semi-dynamical systems (prefixing with “semi” refers to the fact that the dynamics needs not be invertible). This is confirmed by the asymptotic dynamics of SP-RNs, which is independent of the input (Figure 5B). The dynamics within this dynamic regime follows the minimal code. The minimal code manifests itself through a period-4 periodic attractor which corresponds, in the case of Markov-85 input, to the most probable transition in the input space

$A \rightarrow B \rightarrow C \rightarrow D \rightarrow \dots$. This observation confirms the fact that STDP allows the system to learn the basic structure of its input.

SIP-RNs exhibits similar dynamics at the end of the plasticity phase (Figure 5C). However, as is evident from varying the perturbation parameter for SIP-RNs (Figure 4), the set of initial conditions that constitutes this input-insensitive basin is confined by a distance relation to the neighborhood of the periodic attractor: the probability of being in this basin diminishes the further away the initial conditions are from the input-insensitive periodic attractor.

The increase of performance and the neural bandwidth of SIP-RNs for higher π (Figure 4) shows that outside of the *input-insensitive* dynamic regime there exists a different basin of attraction. Within this basin, the network is sensitive to input, and computations are possible. The observation that π has no effect on IP-RNs and that they show intermediate performance and mutual information suggests that they are dominated by a dynamic regime with intermediate input-sensitivity. It also confirms that intrinsic plasticity is responsible for the emergence of the *input-sensitive* dynamic regime in SIP-RNs.

Volumes of Representation

Now that the dynamic regimes of trained networks with the three combinations of synaptic and intrinsic plasticity are identified, we next move to formulating the notion of representations inside the input-sensitive dynamic regime. Developing such a notion allows linking the theory of nonautonomous dynamical systems to a theory of spatiotemporal computations. To this purpose, we coin the term *volumes of representation*, which is a concept that describes the response of a nonautonomous dynamical system in respect to its drive. The volume of representation of some input sequence within some dynamic regime is the set of network states that are *accessible* through exciting the network with the corresponding input sequence, starting from all network states in this dynamic regime as initial conditions (Definition 10). The *order* of a volume is defined by the length of the input sequence it represents. We also introduce the *volumes' inclusion property* which hierarchically links the system's response to spatiotemporal input sequences to their sub-sequences.

To visualize a network's volumes of representation, we sample the network's response. We do this because the size of the state space and the input-sensitive dynamic regime is too large, making a complete coverage impossible. Also, since volumes of representation can have complicated shapes in both the full and reduced state space, we approximate these volumes with ellipsoids.

Figure 5D provides such an approximation to the volumes of representation of order-1. The sample is a single sequence of 10000 Markov-85 inputs to a SIP-RN. Each volume is replaced by an ellipsoid. The center of this ellipsoid is the coordinates' average of the visited network states in the principal components space. Each of its semi-axes has a length that is the standard deviation from the mean of the corresponding coordinate. Also, according to the *volumes' inclusion property*, stated formally in the Methods section, a volume of representation of order-1 of some input p includes all volumes of order-2 for sequences whose most recent input is p . As such, Figure 5E, that depicts a similar approximation to all volumes of order-2, is also a better approximation to volumes of order-1. In Figure 5E, each order-1 volume consists of four order-2 volumes that are color-coded to match the rougher approximation in Figure 5D. In a supporting figure, we further show that this way of presentation is sufficient, compared to using percentiles of bootstrapped network states (see Figure S1).

The volumes of representation provide a geometric view of spatiotemporal computations as the ability of the recurrent neural network to *represent* in its activity, in other words to *encode*, useful functions of the network's input sequences, and for these representations to be distinguishable and reliable. In the case of the tasks RAND \times 4 and Markov-85, the functions that the network activity represents are the identity, delayed or forecast. As shown in Figure 5D–E, the volumes of representation of SIP-RNs under Markov-85 input exhibit higher *separability*, which explains both their high classification performance and high mutual information. One also notices that the volumes of representation of order-2 that belong to the most probable transitions in the Markov-85 input, e.g., $B \rightarrow C$, are also the most distant from one another (Figure 5E). This results in the most probable transitions to be more easily distinguishable by optimal linear classifiers.

In order to isolate the roles of synaptic and intrinsic plasticity in generating useful representations, we show in Figure 5A the order-1 volumes of representation of an IP-RN in response to Markov-85 input. Compared to the SIP-RN, these volumes are highly overlapping, which explains the lower classification performance. Also, the low mutual information between the network state and the input (Figure 3) can now be explained by various network states belonging to multiple volumes of representation, at once. Also, many network states represent the same single input which is a signature of *redundancy* resulting from IP. These observations point towards STDP being the source of separability of representations in SIP-RNs, in addition to learning the structure of the input through situating the representations of the input's most probable transitions at further distances from one another.

In the case of the task Parity-3, the function that the network activity needs to represent is the sequential *exclusive or* operation over three successive binary inputs. As such, within the input-sensitive dynamic regime, two volumes of representation exists, each encodes one outcome of the nonlinear task Parity-3. According to Definition 10, these volumes are formed from an appropriate union of order-3 volumes of representation of the binary input. We provide an illustration of these two volumes of representation in Figure S2. Here also, STDP provides the separability that allows these representations to be distinguishable, while IP gives the possibility of an input-sensitive and redundant regime to emerge, and, aided by STDP, for the volumes of representation to expand.

Attractor Landscape

The presence of dynamic regimes entails the existence of *attractors*, i.e. limit sets of the dynamics, that apply a pulling force on the dynamical system's activity and dictate its course of flow. In an input-driven dynamical system, attractors are not easily defined as sets of states. Instead, *nonautonomous attractors* are input-dependent moving targets of the dynamics, which adds a temporal aspect to their definition (see Definition 8). As follows, for our nonautonomous dynamical systems theory of spatiotemporal computations to be complete, we link the geometry of the computational entities, i.e. the volumes of representation, to the geometry of the nonautonomous attractors. This allows us to connect the features of the volumes of representation emerging from plasticity, namely, separability and redundancy, to the effects of plasticity on the nonautonomous attractor. To that end, starting from the volumes of representations, we define the *perturbation set* (Definition 10) as a moving source of the neural activity towards its moving target, the nonautonomous attractor. Since the perturbation set changes with time, it is called a *nonautonomous set* (Definition 7). This also applies to nonautonomous attractors. The set of states constituting a nonautonomous set at a fixed time t is called the

set's *t*-fiber. We later show how the *t*-fibers of these nonautonomous sets relate to each other.

In the input-insensitive dynamic regime, the dynamical system behaves as an autonomous dynamical system, and so does its attractor, which is the period-4 attractor in Figure 5B–C. In addition, the existence of a *nonautonomous basin of attraction* (Definition 9), that constitutes the input-sensitive dynamic regime in SIP-RNs, necessitates the existence of a *nonautonomous attractor*.

It is not possible to fully identify the nonautonomous attractor by looking into the nonautonomous dynamics. This is because the attractor is not fixed in space and because the dynamics almost never converges to it. However, we prove in Theorem 11.1 that in an input-driven discrete-time dynamical system, and within a basin of attraction, the nonautonomous attractor is a subset of the basin's perturbation set, and that the *t*-fibers of a nonautonomous attractor are subsets of the *t*-fibers of the perturbation set. Given this result, the location of the nonautonomous attractor within the state space of the network can be approximated by the perturbation set. The perturbation set summarizes how the network activity passes from one volume of representation to another, at every time step, according to the input's transition statistics. We replace the time dimension in Figure 5D by arrows that correspond to the transitions in Markov-85 input. The volume of representation visited at time *t* is the volume corresponding to the input at that time, and it forms the *t*-fiber of the perturbation set.

Instead of defining the asymptotic dynamics of the model neural network within the input-sensitive basin of attraction by a single nonautonomous attractor with different *t*-fibers, we can define it by multiple autonomous attractors, each belonging to a particular input. According to Theorem 11.2, within the input-sensitive basin of attraction, there exists for each input *p*, an autonomous attractor (Definition 4) of the autonomous semi-dynamical system defined by *p*. The theorem also shows that this attractor is a subset of the volume of representation of *p*. Theorem 11.3 further shows that the basin of attraction of the autonomous attractor is also the input-sensitive basin. Accordingly, the network dynamics undergoes a *bifurcation* at each time step the input changes its identity. A bifurcation is a change in the topological properties of invariant sets, such as attractors. We observe bifurcations in the input-sensitive regime of kWTA networks. The topological property undergoing the change is the loss of stability of the periodic attractor associated with an input $p(t-1)$, and the appearance of an attractor with a different period and location that is associated with the input $p(t)$.

Figure 5F shows the autonomous periodic attractors associated with each Markov-85 input within the input-sensitive basin of attraction of a SIP-RN. Each of these attractors is also a *t*-fiber of the nonautonomous input-sensitive attractor. While these autonomous attractors are depicted in one state space, overlaying them in a single plot serves only in illustrating the geometric relations between them. In reality, these attractors do not *coexist*. Each autonomous attractor appears in the phase space of the network when its associated input drives the network, and the attractor from the previous time step disappears.

The geometry of the nonautonomous attractor within an input-sensitive dynamic regime is very important regarding spatiotemporal computations. In fact, computations are completely defined according to the relative positions of the nonautonomous attractor's *t*-fibers to one another, and to the volumes of representation. An attractor consists of limit points of a basin of attraction. Thus, it exerts a pulling force on the network states that define the volumes of representation. So, if the *t*-fibers of a nonautonomous attractor are close to one another in the state

space of the network, different volumes will be overlapping and computations will be difficult to carry through. Such is the case in IP-RNs. On the other hand, distant *t*-fibers of the nonautonomous attractor result in separate volumes of representation and better spatiotemporal computations, which is the case in SIP-RNs (Figure 5D–F). Also, the number of states comprising the *t*-fibers of the nonautonomous attractor effects the redundancy of representations. As intrinsic plasticity increases the number of states of these *t*-fibers, the perturbation set becomes more redundant. Given the above, while the perturbation set contains the nonautonomous attractor, it is the attractor that defines how the perturbation set, and as a consequence the volumes of representation, extends in space.

For a correct characterization of spatiotemporal computations according to the geometry of the nonautonomous attractor and function representations, we borrow the concept of *meta-transients* [44]. A transient activity of an autonomous (semi-)dynamical system is the trajectory its dynamics follows as it approaches a fixed attractor. Alternatively, an attractor of an input-driven dynamical system changes constantly. This leads the trajectory pursued by the dynamics to switch its course, so as to keep track of its moving target. Such an input-dependent trajectory is termed a meta-transient. When the input changes, the meta-transient passes from one volume of representation to another, i.e. the dynamics bifurcates and the meta-transient approaches the vertexes of the current attractor, while being repelled from the others that are now unstable. It is in this geometric relation to the different attractors (or *t*-fibers) that computation resides. In fact, as a proof of principle, the autonomous attractors of SIP-RNs were allocated. This was done by clamping each input for a sufficient time until the dynamics converges to that input's periodic attractor. Then, optimal linear classifiers were fitted to perform the three computational tasks. As training data, the Hamming distances between the meta-transient and the vertexes of these periodic attractors were used. Figure S3 shows the performance resulting from this computational procedure, which outperforms both SP-RNs and IP-RNs. While the performance is far from what is achieved directly from the activity of SIP-RNs, especially in the nonlinear task Parity-3, it is important to note that distance is a very rough compression of the geometric relations between the meta-transient and the autonomous attractors. For instance, distance does not allow the distinction between network states that are symmetrical in relation to the autonomous attractors.

Emergence of Computation

We now outline how the interaction of homeostatic and synaptic plasticity gives rise to spatiotemporal computations through developing useful representations. To this end, we combine the analysis of dynamic regimes, volumes of representation, and autonomous and nonautonomous attractors (Figure 5) with the informational-theoretic intuitions regarding the evolution of the neural code (Figure 3).

At the beginning of the plasticity phase, STDP has the upper hand and it generates a minimal code of the input. This is evident from the 2 bits network state entropy (Figure 3A) and the close to zero mutual information with input (Figure 3B) at the beginning of the plasticity phase of SIP-RNs. The minimal code captures, through an input-insensitive periodic attractor, the most probable transitions in the input (Figure 5B). Another feature of the input-insensitive periodic attractor is the high separability of its vertexes in the state space of the SIP-RN.

At the same, IP time succeeds in reducing the excitability thresholds of some neurons, such that more network states become accessible at the vicinity of the vertexes of the input-insensitive

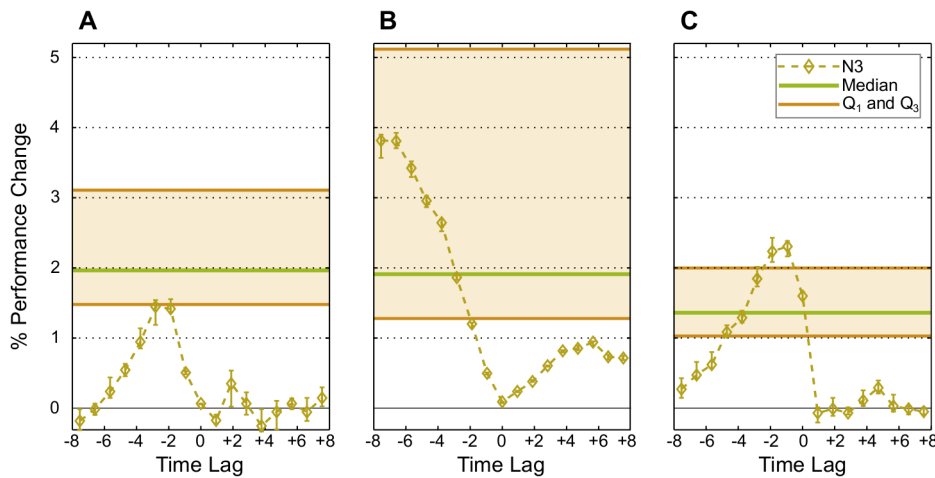


Figure 6. Noise-robustness is achieved through the interaction of synaptic and intrinsic plasticity. Bootstrapped median relative change from the noiseless performance of 100 networks trained with both STDP and IP on (A) the memory task $\text{RAND} \times 4$, (B) the prediction task Markov-85 , and (C) the nonlinear task Parity-3 . High perturbation of $\pi=12$ is applied at the end of the plasticity phase. Error bars correspond to the 25th and the 75th percentiles. Noise level $N3=3\%$ is the probability of a bit flip in the network state, that is, the probability of one of the k spiking neurons at time step t to become silent, while a silent neuron fires instead. The shaded area indicates the ratio of noisy spikes which is measured in comparison to the noiseless SIP-RNs . The green line indicates the median and the orange lines the 25th and the 75th percentiles of the noisy spikes ratio. doi:10.1371/journal.pcbi.1003512.g006

attractor: entropy increases alongside the potential for redundancy. STDP concurrently assigns these network states to the inputs that induce them: mutual information and redundancy increase. This incremental process manifests dynamically in the appearance of the input-sensitive basin of attraction, and the associated appearance and expansion of volumes of representation (Figure 5D–E). Due to the highly separate vertexes of the input-insensitive attractor and the neighborhood relations of the volumes with these vertexes, the volumes of representation are highly separate. This shows that the input-insensitive dynamics is a necessary prerequisite for the emergence of spatiotemporal computations, as it sets the stage for the appearance of separate representations that also carry the structure of the input.

The emerging dynamics can also be viewed through formulating the SIP-RN during the plasticity phase, as an input-driven dynamical system parametrized by the weights and the excitability thresholds. Through varying the parameters of the system with STDP and IP, the dynamics at some point in the parameters space bifurcates from one stable dynamics, the input-insensitive dynamics, to two stable dynamics with the appearance of the input-sensitive attractor in whose basin computations are realizable. This also applies to each member of the family of semi-dynamical systems with the appearance of new dynamics and the associated new periodic attractor (Figure 5F).

Noise-Robustness

Equipped with different vantage points to describe the information processing properties of plastic recurrent neural networks, we now turn to ask a central question: what does an information processing system like the brain require in order to be noise-robust? We state the following hypothesis. Noise-robustness is an effect of the interplay between 1) a *redundant code* that provides multiple possible encodings of an input, and 2) *separability* of representations which allows for a *margin of noise* without obscuring the identity of the input.

The analysis of the neural code (Figure 3) shows how IP increases the potential for redundancy by increasing the neuronal bandwidth. STDP could exploit this potential redundancy by

assigning multiple neurons to the same input. Viewing the network dynamics in the principal components space, on the other hand, made clear that STDP ensures separability in the volumes of representation (Figure 5D–E). This also suggests that the recurrent network should be more robust to noise, the more recent the decoded input is, as the margin of noise becomes smaller for older inputs. The expansion of volumes of representation in IP-RNs also points towards a higher potential redundancy.

We test the hypothesis and the role of STDP and IP interaction in noise-robustness by injecting nondeterministic noise into the recurrent network. Following the plasticity phase, we deploy a certain rate of random bit flips on the network state that reserves the $k\text{WTA}$ dynamics, i.e. if some neuron is silenced due to noise, another neuron is selected at random and it fires instead. Different networks with different input statistics will amplify the same amount of noise to a varying extent. The shaded area in Figure 6 marks the ratio-of-noisy-spikes range within the network states of 100 recurrent networks. For all tasks and networks, we measured performance of optimal linear classifiers on both the noise-free and noisy network states, and computed the relative change in performance.

We compare the change in performance for each time-lag with the ratio of noisy spikes. To understand how this comparison aids in characterizing noise-robustness, we rely on an example. If 10% of a network's spiking activity has been replaced by noise, spikes being the carriers of information, 10% of the information in the network would be lost. However, if the activity of other neurons within the network is a replica of half the lost spikes, only 5% of the information would be lost, and the performance of the linear classifiers would decrease just as much. Having the change of performance below noise level is evidence of noise-robustness due to redundancy and intrinsic plasticity.

Information carried by the network cannot deteriorate beyond the amount of noise; the ability to perform computations, on the other hand, is another story, since distinguishing between representations is a necessary condition for computation. Noise can lead to an overlap in the volumes of representation, which hinders the information processing capability of the recurrent

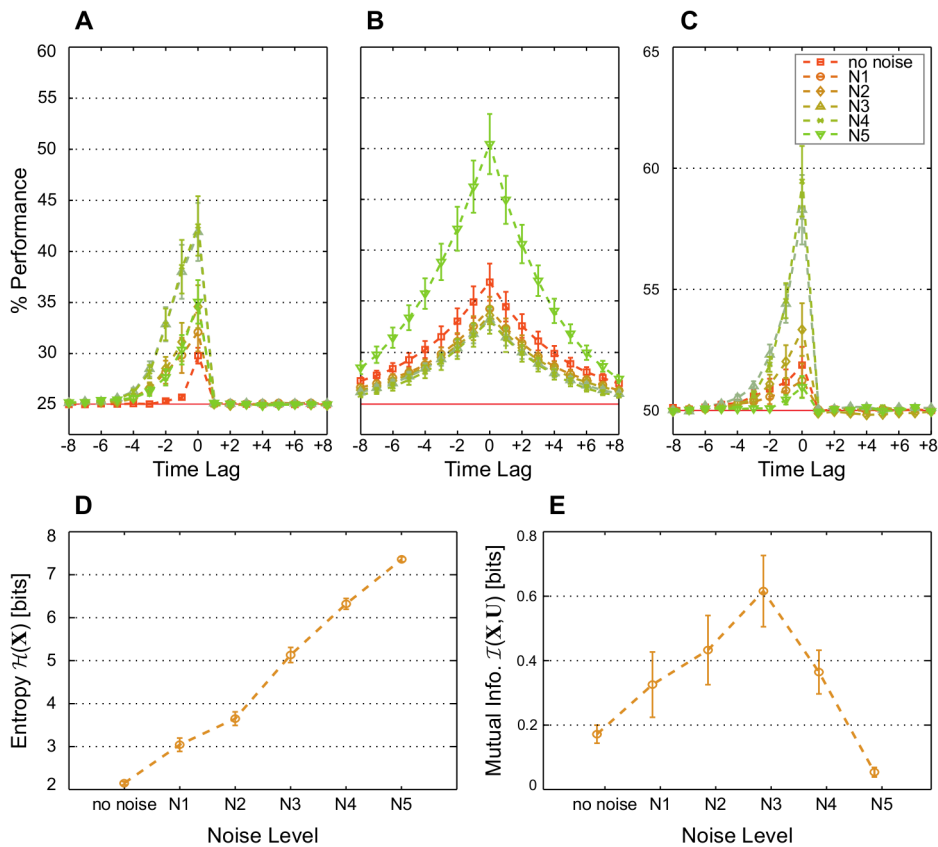


Figure 7. Noise at certain levels is rendered constructive when synaptic and intrinsic plasticity interact. Average classification performance of 100 networks trained with both STDP and IP on (A) the memory task $\text{RAND} \times 4$, (B) the prediction task Markov-85 , and (C) the nonlinear task Parity-3 for increasing levels of noise and no perturbation at the end of the plasticity phase ($\pi=0$). (D) Network state entropy $\mathcal{H}(X)$ and (E) the mutual information with the three most recent $\text{RAND} \times 4$ inputs $\mathcal{I}(U, X)$ at the end of the plasticity phase for different levels of noise. Values are averaged over 50 networks and estimated from 5000 samples for each network. (A–E) Noise levels are applied during the plasticity, training, and testing phases. They indicate the probability of a bit flip in the network state, that is, the probability of one of the k spiking neurons at time step t to become silent, while silent neuron to fire instead. $N1=0.6\%$, $N2=1.2\%$, $N3=3\%$, $N4=6\%$, and $N5=12\%$. Error bars indicate standard error of the mean.

doi:10.1371/journal.pcbi.1003512.g007

neural network, since overlapping representations are indistinguishable and prone to over-fitting by decoders, linear or otherwise. However, when volumes of representation are well separated due to STDP, and redundancy is at play, performance will not exceed the amount of noise in the network: noise-robustness is still achieved.

Figure 6 shows that redundancy and separability are assuring noise-robustness in the three tasks. The effects are the strongest for the task $\text{RAND} \times 4$. The change of performance never exceeds the range of noise for all time-lags. The change of performance on the task Markov-85 remains below the range of noise for few time-lags in the past and it remains within the bounds of the noise range for older stimuli. The networks then are still capable of tolerating noise, while the volumes of representation are becoming more overlapping. The decrease of noise-robustness for larger time-lags in the past confirms our suggestion that volumes of representation become less separate for older inputs. The analysis of order-2 volumes of representation (Figure 5E) also suggests that less probable transitions of the input are more prone to noise. This, however, was not tested. The task Parity-3 is noise-robust for 0-time-lag only and with the change in performance being within the noise range. This is understandable, since for each time-lag, order-3 volumes of representation and the associated volumes of the Parity-3 function should be separate and redundant.

These observations confirm our hypothesis that redundancy and separability are the appropriate ingredients for a noise-robust information processing system, such as our model neural network. These properties being the outcome of STDP's and IP's collaboration, suggest the pivotal role of the interaction between homeostatic and synaptic plasticity for combating noise.

Constructive Role of Noise

Now that we have demonstrated the contributions of STDP and IP in combating noise, we turn to investigating noise's beneficial role. We have seen that perturbation at the end of the plasticity phase provides a solution to the network being trapped in an input-insensitive regime. Besides viewing perturbation as a form of one-shot strong noise, which is, biologically speaking, an unnatural phenomenon, what effect would a perpetual small amount of noise have on the dynamics of the recurrent neural network?

We again deploy a certain rate of random bit flips on the network state that reserves the $k\text{WTA}$ dynamics. Unlike the previous section, we do not restrict noise to the training and testing phase, but apply it also during the plasticity phase. We also do not reset the network activity after the plasticity phase, i.e. the perturbation parameter π is set to 0.

Figure 7A–C compares the performance of optimal linear classifiers on the three tasks for different levels of noise. For all

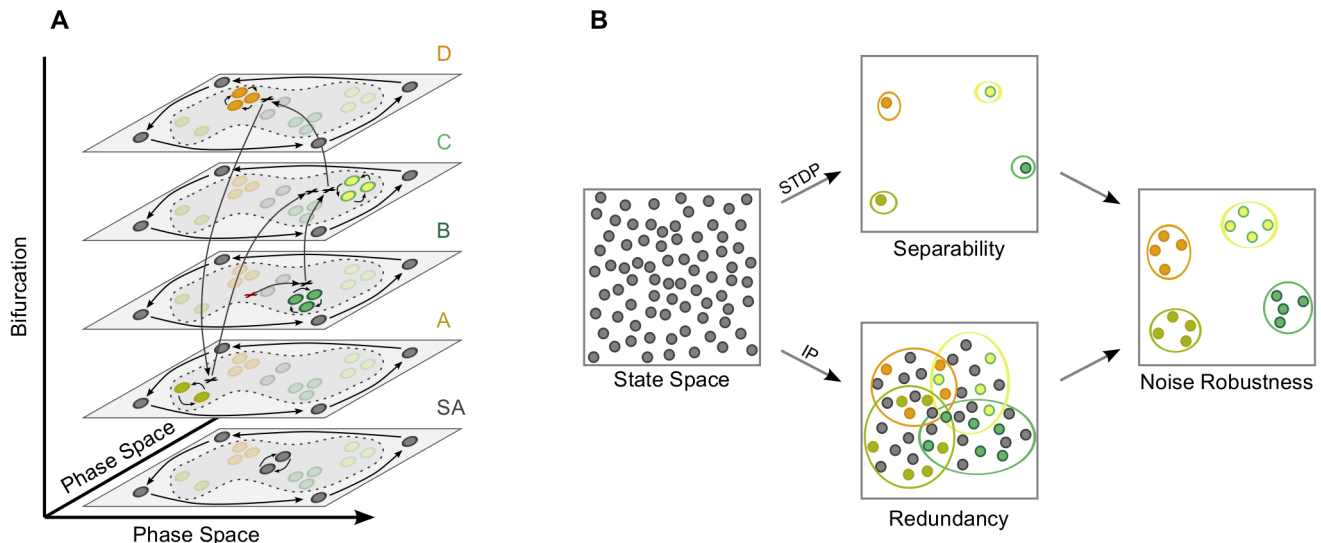


Figure 8. Schematics of the driven dynamics of networks endowed by synaptic and homeostatic plasticity, and the emergence of noise-robust spatiotemporal computations. (A) The dynamics of a recurrent network that is trained by homeostatic and synaptic plasticity and driven by a Markovian input. Each layer corresponds to one input. The layer illustrates a two-dimensional projection of the phase space of the autonomous (semi-)dynamical system associated with that input. A layer that corresponds to the spontaneous activity (SA) is added for completeness. Due to the interaction of synaptic and homeostatic plasticity, each of these (semi-)dynamical systems has two dynamic regimes: an input-insensitive dynamic regime that is shared by all the layers and that captures the temporal structure of the input, and an input-sensitive dynamic regime that contains a single periodic attractor. The input-sensitive attractor depends on the layer and is close to one of the vertexes of the input-insensitive attractor. The network is excited by the exemplary input sequence $B \rightarrow C \rightarrow D \rightarrow A \rightarrow C$. The red cross refers to the initial conditions that are chosen within the input-sensitive dynamics. Given the input sequence, the network dynamics follows the meta-transient that is illustrated by the arrows between the different layers. For instance, when the network is excited by the input B , the network activity approaches the B -attractor within the corresponding layer. When C follows, a bifurcation occurs, where the B -attractor becomes unstable and the C -attractor becomes stable. The meta-transient approaches the C -attractor from the direction of the unstable B -attractor. When C is preceded by the less common input A , the C -attractor is approached differently, such that the distance to it is bigger than in the case of the most common transition $B \rightarrow C$. (B) Noise-robust computations are a result of the interaction between synaptic and homeostatic intrinsic plasticity. Synaptic plasticity leads to high separability and intrinsic plasticity to redundancy. These effects lead to a neural code that allows a higher margin of noise and alternative representations of computations, thus facilitating noise-robustness.

doi:10.1371/journal.pcbi.1003512.g008

tasks, some levels of noise resulted in a significantly higher average performance than the noiseless case. The task `Markov-85` had the highest average performance at the largest level of noise, while the tasks `RAND x 4` and `Parity-3`, where the input was uniformly random, had the highest performance at the third and fourth levels of noise, and the average performance dropped substantially at the fifth level of noise. In all tasks, performance was far off the levels it reached in the noiseless case (Figure 2).

Information-theoretical quantities are again measured on networks with `RAND x 4` input. As expected, the network state entropy increases monotonically with noise (Figure 7D). Mutual information, on the other hand, starts dropping for noise larger than the third level (Figure 7E). This is also expected from the change of performance (Figure 7A). Noise then appears to provide, in some of the `SIP-RNs`, the necessary means to escape the input-insensitive dynamics. At some levels, however, the network activity becomes dominated by noise beyond the compensatory effects of redundancy and separability achieved through plasticity. In addition, more unstructured noise during the plasticity phase delays the creation and expansion of useful volumes of representation, thereby hindering computations further.

Discussion

We demonstrated how the interaction of synaptic learning and homeostatic regulation boosts memory capacity of recurrent neural networks, allows them to discover regularities in the input stream, and enhances nonlinear computations. We provided a

geometric interpretation of the emergence of these spatiotemporal computations through analyzing the driven dynamic response of the recurrent neural network. We view computations as a geometric relationship between *representations* of functions over stimuli, representations that consist of network states, and the asymptotic dynamics of the network, i.e. attractors. Accordingly, Figure 8A shows a possible driven-dynamics viewpoint on computation, which is the following. As the stimulus changes, a bifurcation occurs where the current attractor of the network becomes unstable, while another stabilizes according to the current stimulus. That leads the network dynamics to change its course towards the new stable region, or attractor, of the state space, and away from the previous attractors that are all unstable. As such, this path of the network activity, i.e. the meta-transient [44], is defined by both the stimulus sequence and the locations of the network's attractors. Together, they lead the meta-transient to pass through particular representations which encode computations. An equivalent alternative to the *chain of bifurcations* between autonomous attractors is that of a single *nonautonomous attractor* that behaves as a stimulus-dependent moving target of the dynamics.

We showed that a successful implementation of these spatiotemporal computations requires the interaction of synaptic and homeostatic intrinsic plasticity which generates *useful representations* in the dynamics of excitable cortical networks. Figure 8 schematically illustrates the stimulus-driven dynamical viewpoint of spatiotemporal computations and the effects of plasticity. Synaptic plasticity produces stimulus-insensitive dynamics that captures the temporal structure of the input. Intrinsic plasticity increases the

neuronal bandwidth by increasing sensitivity to stimuli, which reduces the dominance of the stimulus-insensitive dynamics. This, in combination with synaptic plasticity, generates stimulus-sensitive attractors and *redundant* representations around them. These stimulus-sensitive components are pulled apart by the stimulus-insensitive dynamics, so that the structure of the input is preserved, and the *separability* of representations is higher and computations are realizable.

We pointed out throughout the text that computation is an *emergent* property of the recurrent network, and that it cannot be fully understood from the individual contribution of the parts, be it neurons or plasticity mechanisms. It might appear contradictory to that statement that the analysis was often concerned with the isolated role of each single plasticity mechanism. However, the quantitative assessments of computations point back to the emergent and collective aspect of computation. Namely, measured on SIP-RNs, neither performance of linear classifiers nor mutual information with input can be accounted for by a linear relationship between the respective quantities measured on SP-RNs and IP-RNs. In fact, the performance of networks where the recurrent weights and firing thresholds are adapted separately, and then combined following the plasticity phase, is far less than the performance of SIP-RNs, where intrinsic and synaptic plasticity are mutually active (see Figure S4). This further consolidates the claim that computations in SIP-RNs *emerge* from the *interaction* of STDP and IP, and *not* from their isolated contributions. It also points back to the formation of separate and redundant representations from the continuous interplay of these two mechanisms.

We also illustrated the combined role of synaptic and homeostatic intrinsic plasticity in creating noise-robust encoding through the generation of a redundant neural code. Many studies have investigated the redundant nature of neural information transmission in many cortical regions, and have justified this expensive allocation of neural resources by redundancy serving as an error-correction strategy that provides neural assemblies with the capacity to average out noise [10,47–50]. Tkačik and colleagues have shown that in the presence of noise, a maximum entropy model of the retina increases redundancy for higher noise levels. A side effect of their model is that stimulus representations become highly separate, which increases the tolerance margin of noise and enhances information transmission [51]. Our model was able, through *local* plasticity mechanisms, to capture both of these properties, achieved in [51] through optimality principles, and to lead to a noise-robust population code (Figure 8B). Namely, synaptic plasticity enhances the separability of representations through the pulling force of the input-insensitive attractor, while intrinsic plasticity perturbs the network states and increases redundancy when interacting with synaptic plasticity, which allows for alternative representations of similar input sequences. Another point of similarity with the model of Tkačik and colleagues [51] and with empirical findings [52,53] is the remnant fingerprint of the most common stimulus in the network's spontaneous activity, which manifests in our model neural network in the stimulus-insensitive dynamics (Figure 5B–C).

In addition to combating noise, our model explores a potential benefit from its presence. We pointed out the necessity of the stimulus-insensitive dynamics for the emergence of computation in the model neural network. The stimulus-insensitive attractor provides the baseline dynamics for the appearance of highly separate representations, and thus, the excitable dynamics necessary for computations. Getting from the input-insensitive regime to the excitable one depended, however, on the *ad hoc*

reinitialization of the network activity at the end of the plasticity phase. Noise provides an alternative. During the plasticity phase, noise shallows the boundaries between the two basins of attraction, which reduces the dominance of the stimulus-insensitive attractor. After the plasticity phase, noise supplies the small perturbations needed to get the network activity to the sensitive dynamics where computations are possible. This solution, in comparison to reinitializing the network activity, is more inferior, specifically because noise also delays the learning of representations. We postulate that another homeostatic plasticity mechanism, *synaptic scaling*, might contribute to the shallowing of the attractor boundary by constraining the strength of synapse bundles between neural subpopulations (e.g., between RF_A and RF_C). For instance, synaptic scaling was necessary for implementing spatiotemporal computations in *self-organizing recurrent networks* (SORN) [28], but no analysis of the dynamics of these networks was done. Testing this hypothesis is, however, beyond the scope of this work.

It is also tempting to connect the topology of the attractor landscape of SIP-RNs to neuropathology and to a model by Pfister and Tass [54]. They suggest that two stable regimes of recurrent network activity, a synchronous pathological regime and an asynchronous healthy regime, coexist, and that their coexistence is a necessary condition for the functioning of a model of deep brain stimulation. In their model, the stimulation of the recurrent network destabilizes the synchronous dynamics through inducing STDP. The destabilization drives the network activity towards the healthy asynchronous basin of attraction. By eliminating the stimulation, the energy hill between the two dynamic regimes rises again and the network remains in the healthy dynamics. Our study has shown how these two coexisting dynamic regimes and their associated forms of activity might come into being through neuronal plasticity. We also suggested noise as a possible mechanism for avoiding the unhealthy dynamics. Further analysis is necessary to investigate how the interaction between noise and different plasticity mechanisms might contribute to our understanding of neurological disorders.

Our analysis of spatiotemporal computations was restricted to Markovian dependencies in the temporal structure of the stimulus or to no dependencies at all. This is often not the case in natural stimuli faced by animals and humans, where the Markov property does not always hold. Lazar et al. have shown that SIP-RNs are capable, to a certain degree, of performing predictions on second-order Markov chains [27]. However, optimal encoding of non-Markovian stimuli and performing computations over them require forms of spike-timing-dependent plasticity that are less myopic to the temporal dependencies than what we considered in this work (Figure 1B). For instance, Brea and colleagues have shown that storing and reproducing a non-Markovian sequence in a recurrent neural network require a nonlocal form of STDP with more complex temporal dependencies between pre- and post-synaptic spikes [55]. While their model was not concerned with carrying through spatiotemporal computations of the kind we presented here, it successfully reproduced the stored non-Markovian input in the spontaneous activity of the neural network. This refers to a point of similarity to the simpler case we presented here, where Markovian input was stored and recalled in the spontaneous activity of the input-insensitive dynamics. In any case, while spatiotemporal computations over non-Markovian stimuli and the necessarily more complex plasticity mechanisms that lead to their emergence, are not considered here, we view the concepts and methodology developed above as a general framework for future studies.

In this article, we provided a first analysis of the combined role of synaptic and intrinsic plasticity on the emergent dynamics of

recurrent neural networks subject to input. We redefined computations in relation to these emergent dynamics and related that to properties of the neural code. We also considered how the neural dynamics interact with noise, both as a nuisance to combat, and as a driving force towards healthy neural activity. The model we used is simplified, however, both in network architecture and plasticity mechanisms. While this simplification is necessary for mathematical convenience, biology never cares for formal abstractions, for the brain is a complex information processing system that is rich with a variety of neuronal morphologies and functions. The plastic changes the brain undergoes are neither confined to the two mechanisms we dealt with here, nor are they uniform across different regions. On the other hand, mathematical formalization of computation and adaptability allows the identification of unifying principles in computational biology, in general, and neural computations, in particular. We intended the current article as a step in that direction.

Methods

The setup on which we assessed spatiotemporal computations in recurrent neural networks is partially inspired by the theory of *reservoir computing* (RC) [3,29,30]. However, as shown in the Results section, our analysis is independent of the RC paradigm, as it is concerned with the effects of plasticity on the recurrent network, and optimal linear classifiers are only used as one possible probe to quantify these effects. We present in this section the *recurrent network* (RN) architecture and the plasticity mechanisms active in shaping the neural response. We follow by introducing the computational tasks and justifying their selection. We then specify the simulation conditions and the training of optimal linear classifiers, followed by demonstrating how information-theoretical quantities are estimated. We finally lay down the mathematical formalization of the autonomous, input-driven, and input-insensitive dynamics of the recurrent network: We adapt Definitions 2, 4, 6–8 from [31] to the special case of discrete-time dynamics [32], which is the case that concerns the current article. We contribute the new concepts of volumes of representation notation and purposes.

Network Architecture

In this paper, the model recurrent network is of the *k-Winner-Take-All* (kWTA) type [27] that consists of n memoryless binary neurons from which only k neurons are active. The discrete-time dynamics of the recurrent network at each time step $t \in \mathbb{Z}^+$ is given by

$$x(t+1) = f(w \cdot x(t) - \theta + \delta(t)), \quad (1)$$

where $x \in \mathbb{R}^n$ is the network state. The nonlinear function f sets the k units with the highest activities to 1 (spiking), and the rest to 0 (silent). As such, the population firing rate is held constant at k , and there is no need to introduce inhibitory neurons to balance excitation and inhibition. Recurrent synaptic efficacy is defined by the weight matrix $w \in [0,1]^{n \times n}$ with w_{ij} being the efficacy of the synapse connecting neuron j to neuron i . Self-coupling is avoided by setting diagonal elements w_{ii} to 0. $\theta \in \mathbb{R}^n$ defines neuronal firing thresholds that modulate the neurons' resistance to firing, and hence, their excitability. $\delta \in \mathbb{R}^n$ is the external drive whose dynamics depends on the task performed.

More formally, the set of possible network states is a metric space:

Definition 1. Given the set $Y = \mathcal{B}^n = \{0,1\}^n$ of all binary vectors of size n , we define the *Hamming metric* by the function:

$$d : Y \times Y \rightarrow \mathbb{Z}^+ \cap [0,n] : d(y_1, y_2) = \sum_{i=1}^n |y_1^{(i)} - y_2^{(i)}|$$

According to this metric, the distance between two vectors of Y is the number of bits at which these two vectors differ. The Hamming metric is a proper metric on strings of fixed length which is the case for Y . The pair (Y, d) then forms a *metric space*. It is also equivalent to the L_1 norm on the set Y , which allows us to define the *Hamming length* of a vector $y \in Y$ as the Hamming distance between y and the 0-vector, i.e. $|y|_d = d(y, 0)$.

Given the kWTA dynamics (see Equation 1), the network activity is restricted to the set:

$$X = \mathcal{B}_k^n = \{x \in Y : |x|_d = k\} \quad (2)$$

Since $X \subset Y$, the pair (X, d) forms a metric space as well. Distances between subsets of X can be measured using the Hausdorff metric, which we also denote d .

Plasticity Mechanisms

We are concerned with the interplay of two forms of plasticity in enhancing the computational capability of the model recurrent network.

Spike-timing-dependent synaptic plasticity (STDP) is a set of Hebbian/anti-Hebbian learning rules, where synaptic efficacy is modified according to the relative firing time between pre- and post-synaptic neurons [56]. We adapted a simple causal STDP learning rule by which a synapse is potentiated whenever the pre-synaptic neuron fires one time step before the post-synaptic neuron, and is depressed when a post-synaptic spike precedes a pre-synaptic spike by one time step:

$$\Delta w_{ij}(t+1) = \eta_{sp} (x_j(t) \cdot x_i(t+1) - x_i(t) \cdot x_j(t+1)), \quad (3)$$

where η_{sp} is the synaptic plasticity learning rate set to 0.001. To prevent weights from switching signs or growing uncontrollably, we enforce hard bounds such that the weights remain within the interval $[0, 1]$.

Competition between synapses due to STDP leads to neurons with synapses that won the competition to fire consistently and those who lost the competition to be constantly silent [57]. To counteract this pathological state, the time-averaged firing rate for a neuron is modulated through homeostatic modification of its excitability threshold using **intrinsic plasticity** (IP) [6,7]:

$$\Delta \theta_i(t+1) = \eta_{ip} (x_i(t+1) - k/n), \quad (4)$$

where η_{ip} is the intrinsic plasticity learning rate set to 0.001. This rule uses subtractive normalization to pull the time-averaged firing rate of each neuron closer to the population firing rate k .

Computational Tasks

Neural circuits in different brain regions adapt to best serve the region's functional purpose. To that end, we constructed three tasks, each of which resembles in spirit the demands of one such canonical function. We then, under the stimulation conditions of each task, compared the performance, information content, and dynamical response of networks optimized by combining both STDP and IP with networks that are optimized by STDP alone or IP alone.

In all tasks, the network is subject to perturbation by a set of inputs P . The receptive fields of non-overlapping subsets of neurons $x^{(p)}$ are tuned exclusively to each input $p \in P$. As such, each input p has its corresponding receptive field $x^{(p)} = RF_p$ in the recurrent neural network. When an input p drives the network, all neurons $x^{(p)}$ receive a positive drive $d = 0.25$, while the rest $x \setminus x^{(p)}$ receive none. Readouts are trained on the current network state $x(t)$ to compute a function over input sequences $u_{\tau_1, \tau_2}(t) = \langle p(t + \tau_1), \dots, p(t + \tau_2) \rangle$, τ_1 and τ_2 being time-lags at which target inputs are applied where positive lags corresponds to future inputs and negative lags to past ones. We restrict time-lags τ to the range $\mathbb{Z} \cap [-8, 8]$.

In a first task, `RAND x 4`, we assessed the capacity of the network to retain memory of past stimuli within its activity. The recurrent network is driven by four randomly drawn inputs $P = \{A, B, C, D\}$. The receptive field of each input consists of 15 neurons, and one optimal linear classifier $\mathcal{O}^{(p, \tau)}$ is trained for each input/time-lag pair, i.e. $\mathcal{O}^{(p, \tau)}$ fires when $p(t + \tau) = p$ and is silent otherwise.

The second task, `Markov-85`, explores the ability of the recurrent network to discover temporal regularities in its input. The recurrent network receives one of four possible inputs $P = \{A, B, C, D\}$ generated from a Markov chain with 85% probability for A to be followed by B , B followed by C , C followed by D , and D followed by A . All other transitions occur with a 5% probability. Again, the receptive field of each input consists of 15 neurons, and one optimal linear classifier $\mathcal{O}^{(p, \tau)}$ is trained for each input/time-lag pair.

With the third task, `Parity-3`, we exploit the nonlinear expansion provided by the recurrent neural network. Here, the network is subject to binary input $P = \{0, 1\}$, where each symbol has a receptive field of 40 neurons. The task is to identify the parity of a sequence of three successive inputs. This means that given an input sequence $u_{(\tau-2)\tau}(t) = \langle p(t + \tau - 2), p(t + \tau - 1), p(t + \tau) \rangle$, an optimal linear classifier $\mathcal{O}^{(1, \tau)}$ fires when $g(u_{(\tau-2)\tau}(t)) = p(t + \tau - 2) \oplus p(t + \tau - 1) \oplus p(t + \tau) = 1$, and is silent otherwise. \oplus is the nonlinear *exclusive or* binary operation.

Even though every task used here is very much simplified compared to stimuli usually processed by neural systems, we would still like to link the basic properties of every task presented here to a realistic case processed by a human or an animal. The property of the memory task `RAND x 4` that we want to emphasize is that a neural system must be able to process rapidly changing stimuli that are only shortly presented. That property is partly reminiscent of retinal input, which is rather stationary during moments of fixation, and rapidly changing due to saccadic eye movements. However, it needs to be noted that saccadic eye movements might be difficult to predict and may appear rather random, but are very likely structured and stimulus-dependent. This motivated the prediction task `Markov-85` that models temporally structured and rapidly changing sensory input that is shortly presented. Such input could either be generated by retinal input and saccadic eye movements, or by the whisking behavior and the produced neural activity in the barrel cortex of a mouse. In addition, nonlinearities are prevailing in natural stimuli, and to highlight the necessity of processing these stimuli, we used the nonlinear task `Parity-3`. Such computational demands can be easily motivated by occlusion in vision, where pixel intensities do not sum up linearly at points where one object occludes another in the visual field. Again, we stress that none of these tasks is a good model of real processing in neural systems in nature. However, each is sharing individual aspects that are motivated by real life examples.

Simulation Conditions

In order to isolate the role of STDP and IP in shaping the computational and information processing properties of the recurrent network, we compared networks trained by both STDP and IP, with networks that are trained by STDP alone or IP alone.

Throughout all experiments, we trained networks of $n = 100$ neurons on either the STDP+IP condition, the STDP condition, or the IP condition for a *plasticity phase* of t_{pl} time steps. For convenience, we call a *recurrent network* trained with both *synaptic* and *intrinsic* plasticity SIP-RN. In contrast, we name a recurrent network that learned with a single plasticity mechanism either SP-RN or IP-RN. k is set to 12, the initial weights are chosen uniformly on the interval $[0, 0.1]$ with 10% connectivity probability, and thresholds are drawn from a Gaussian distribution with 0 mean and 0.1 standard deviation. Under the IP condition, to assure that weights' distribution is not different from conditions where STDP modifies the synaptic efficacies w , a *pre-plasticity phase* of similar length to the plasticity phase precedes the latter, where both STDP and IP are active. Afterwards, the weights structure is destroyed by random shuffling and the plasticity phase starts where IP is turned on.

In all experiments where the performance of optimal linear classifiers is estimated, the plasticity phase was $t_{pl} = 20000$ time steps long. Afterwards, weights and thresholds are held fixed, the network state is *reset* to a random initial state, and the *training phase* starts where linear classifiers are trained using linear regression on $t_{tr} = 5000$ time steps, followed by a *testing phase* of performance for another $t_{ts} = 5000$ time steps.

Post-Plasticity Perturbation

At the beginning of the training phase, the network state is reset to a random initial state. If the network dynamics is multistable, this resetting could bring it to a different regime than where the network was at the end of the plasticity phase. To test this possibility systematically, we perform the following post-plasticity perturbation.

Given some perturbation parameter $\pi \in \mathbb{Z} \cap [0, k]$. We assume the network state at the end of the plasticity phase is $x(t_{ps})$. Instead of randomly choosing the initial network state for the training phase, we choose a network state x_π such that the condition $d(x_\pi, x(t_{ps})) = 2\pi$ holds. To satisfy this condition, x_π is chosen as follows. In the network state $x(t_{ps})$, π firing neurons and π silent neurons are randomly selected. The π firing neurons are then silenced and the π silent neurons are set to firing.

Output Weights and Performance

According to the RC paradigm, an input signal undergoes a nonlinear feature expansion by projecting into a recurrent neural network of nonlinear units. The network recurrency also provides a sustained but damped trace of past inputs (echo state [29] or fading memory [30]) to propagate through the network. The network state is then read out by simple linear units through linear regression.

Following the plasticity phase, the network activity during the training phase

$$X_{tr} = [x^T(t)]_{t_{pl} < t \leq t_{pl} + t_{tr}} \in \{0, 1\}^{t_{tr} \times n} \quad (5)$$

provides the training data for all optimal linear classifiers, where $(\cdot)^T$ denotes matrix transpose. The target signal of output neurons for a particular time-lag τ is clamped in a supervised fashion to

$$\mathcal{O}_{ir}^{(\tau)} = [\mathcal{O}^{p,\tau}(t+\tau)]_{t_{pl}+t_{tr} < t \leq t_{pl}+t_{tr}} \in \{0,1\}^{t_{tr} \times |g|}, \quad (6)$$

where $|g|$ depends on the task and is the cardinality of the set of possible values which the target signal can take. $|g|$ equals $|P|$ for the tasks `RAND × 4` and `MARKOV-85`. Output weights w^o for each time-lag are then computed using linear regression through ordinary least squares

$$w^{(\tau)} = X_{ir}^\dagger \cdot \mathcal{O}_{ir}^{(\tau)} = (X_{ir}^T X_{ir})^{-1} X_{ir}^T \cdot \mathcal{O}_{ir}^{(\tau)} \in \mathbb{R}^{n \times |g|}, \quad (7)$$

where $(\cdot)^\dagger$ is the *Moore-Penrose pseudoinverse* of a matrix, and $(\cdot)^{-1}$ is the regular inverse of square matrices.

These optimal linear classifiers are then validated on the network activity

$$X_{is} = [x^T(t)]_{t_{pl}+t_{tr} < t \leq t_{pl}+t_{tr}+t_{is}} \in \{0,1\}^{t_{is} \times n} \quad (8)$$

during the testing phase. First, a pre-estimate of the target signal is computed for each time-lag:

$$\tilde{\mathcal{O}}^{(\tau)} = X_{is} \cdot w^{(\tau)} \in \mathbb{R}^{t_{is} \times |g|} \quad (9)$$

Only one output neuron fires each time step for each time-lag, and this is specified through winner-take-all on the rows of $\tilde{\mathcal{O}}^{(\tau)}$. This leads to the final estimate $\mathcal{O}_{is}^{(\tau)} \in \{0,1\}^{t_{is} \times |g|}$. The *classification performance* for each time-lag is finally computed as the percentage of correct classifications:

$$\Pi^{(\tau)} = \frac{100}{t_{is}} \cdot \sum_{i=1}^{t_{is}} \sum_{j=1}^{|g|} \left(1 - \frac{|\mathcal{O}_{is}^{(\tau)}(t,i) - \tilde{\mathcal{O}}^{(\tau)}(t,i)|}{2} \right) \quad (10)$$

Computing Entropy and Mutual Information

On multiple occasions, both the self-information capacity of the network state and its dependence on input was measured. Entropy measures self-information capacity which is the expected value of information carried by the network activity X and is given by

$$\mathcal{H}(X) = - \sum_{x \in X} p(x) \cdot \log p(x), \quad (11)$$

where $\log(\cdot)$ is the base-2 logarithm, so that entropy (and mutual information) are measured in bits. Mutual information measures the dependence of the network activity X on a corresponding input sequence U and is given by

$$\mathcal{I}(U, X) = \sum_{u \in U} \sum_{x \in X} p(u, x) \cdot \log \frac{p(u, x)}{p(u) \cdot p(x)} \quad (12)$$

In computing entropy and mutual information, we used the algorithm and code developed in [58] that computes entropy from an adaptive k-nearest-neighbor estimate of probability density

functions. This allows for reliable estimates of these quantities with far fewer samples in comparison to other algorithms. Nevertheless, due to the high number of channels we have (100 neurons), and to truncate unnecessary computation time, samples from the network activity are first transferred to the principal components space, and only components that carry 95% of the information are passed to the mutual information estimator.

We always considered inputs from the task `RAND × 4` and we computed the mutual information between samples of the network state $x(t)$ and the three most recent inputs $u(t) = \langle p(t-2), p(t-1), p(t) \rangle$. We encoded each of the four input symbols $P = \{A, B, C, D\}$ by a 3-bits code $\tilde{P} = \{000, 011, 101, 110\}$ to ensure equal pairwise Hamming distances between symbols. For all cases but one, as few as 5000 samples of the network state $x(t)$ and input sequence u were enough to reliably estimate entropy and mutual information. The exception was computing mutual information between input and `IP-RN` activity, which demanded a higher number of samples (500000 time steps) and very long computation time, as covering 95% of the information required no less than 60 principal components.

Autonomous Dynamics

For a full understanding of the emerging information processing properties of the interaction of synaptic and intrinsic plasticity, it was necessary to rely on and develop concepts from the newly emerging mathematical theory of *nonautonomous dynamical systems* [31,32]. Throughout what follows, the correspondence of the introduced concepts to our model is clarified. First, autonomous dynamics are defined, since they form a special instance of the nonautonomous case.

Definition 2. Let (X, d) be a metric space with a metric d . A *discrete-time semi-dynamical system* is a function $\phi : \mathbb{Z}^+ \times X \rightarrow X$ that satisfies

1. $\phi(0, x) = x \quad \forall x \in X$.
2. $\phi(t_1 + t_2, x) = \phi(t_2, \phi(t_1, x)) \quad \forall x \in X$ and $\forall t_1, t_2 \in \mathbb{Z}^+$.
3. ϕ is continuous.

Equation 1 defines the driven or nonautonomous `kWTA` dynamics. The autonomous alternative is given by the discrete-time difference equation

$$x(t+1) = f^{(0)}(x(t)) = f(w \cdot x(t) - \theta), \quad (13)$$

where f is the `kWTA` nonlinearity defined as above. To relate Equation 13 to Definition 2, the function ϕ (the *solution mapping*) is chosen such that

$$\phi(t, x) = \underbrace{f \circ f \circ \dots \circ f}_{t \text{ times}}(w \cdot x - \theta), \quad (14)$$

where \circ is function composition. For ϕ to be an autonomous semi-dynamical system, it has to satisfy the three conditions of Definition 2. The first two conditions are trivial, as they result from the definition of function composition. We turn to prove the third condition, namely, the continuity of ϕ . We first observe that ϕ is merely the t -fold composition of the function f , and since the composition of continuous functions is continuous, it is sufficient to prove the continuity of f .

Proposition 3. The `kWTA` function $f^{(0)}$ from Equation 13 defined on the metric space (X, d) is continuous, i.e.

$\forall x_1 \in X$ and $\forall \epsilon > 0; \exists \delta > 0$ such that

$\forall x_2 \in X$ with $d(x_1, x_2) < \delta$, $d(f^{(0)}(x_1), f^{(0)}(x_2)) < \epsilon$ holds.

Proof. For all $x_1 \in X$ and all $\epsilon > 0$, we choose $\delta(\epsilon) = 1$. For all $x_2 \in X$, if the Hamming distance $d(x_1, x_2) < 1$, x_1 and x_2 have to be equal, since the kWTA dynamics restricts the distances between any two states to the set $[0, m] \cap \{0, 2, \dots, 2k\}$. As such, since d is a metric, $d(f(x_1), f(x_2)) = 0$, which is always smaller than $\epsilon > 0$. Ergo, f is continuous.

We note that the proof to Proposition 3 becomes trivial if we consider a result from topology which states that any function from a *discrete topological space* to another is continuous. However, the proof is interesting in that it shows that $f^{(0)}$ has a stronger form of continuity, that is, $f^{(0)}$ is *uniformly continuous*, since the proof shows that there exists a *packing radius* $r = \delta = 1 > 0$ such that either $x_1 = x_2$ or $d(x_1, x_2) > r$ for all x_1 and $x_2 \in X$.

With the proof of Proposition 3, we conclude that the kWTA autonomous dynamics in Equation 13 generates a discrete-time semi-dynamical system. A dynamical system is a semi-dynamical system with invertible dynamics, which is not the case for kWTA networks. However, for all intents and purposes, being a semi-dynamical system is sufficient for formalizing the nonautonomous dynamics of the model network.

Autonomous Attractors

Characterizing the computational properties of the model neural network requires defining invariant sets and attractors.

Definition 4. Let $\phi : \mathbb{Z}^+ \times X \rightarrow X$ be a discrete-time semi-dynamical system generated by an autonomous difference equation f on a metric space (X, d) . A subset $\mathcal{A} \subseteq X$ is *invariant* under ϕ and f if $f(\mathcal{A}) = \mathcal{A}$, and is *positively invariant* if $f(\mathcal{A}) \subset \mathcal{A}$. \mathcal{A} is an *attractor* of ϕ if the following conditions hold:

1. \mathcal{A} is invariant under ϕ and f .
2. \mathcal{A} is compact.
3. There exists a neighborhood \mathcal{N}_r of radius $r > 0$ of \mathcal{A} such that $\lim_{t \rightarrow \infty} d(\phi(t, \mathcal{N}_r(\mathcal{A})), \mathcal{A}) = 0$

For the kWTA dynamics, the second condition is assured, since X is discrete and finite, which makes all subsets compact. The third condition assures that no subset of \mathcal{A} satisfies the invariance and compactness conditions. Another important concept is that of a *basin of attraction* which associates each attractor with the region of the state space that converges to that attractor:

Definition 5. Let $\phi : \mathbb{Z}^+ \times X \rightarrow X$ be a discrete-time semi-dynamical system generated by an autonomous difference equation f on a metric space (X, d) . The *basin of attraction* of an attractor \mathcal{A} of ϕ is defined by

$$\mathcal{B}_{\mathcal{A}} = \{x \in X : \lim_{t \rightarrow \infty} d(\phi(t, x), \mathcal{A}) = 0\}$$

Nonautonomous Dynamics

Unlike autonomous (semi-)dynamical systems, the elapsed time is not sufficient to find the solution for nonautonomous dynamics: both the start and end times must be specified. Accordingly, we now define a *discrete-time nonautonomous dynamical system* as a *process*. In what follows, we will make use of the set $\mathbb{Z}_{\geq}^2 = \{(t, t_0) \in \mathbb{Z}^2 : t \geq t_0\}$.

Definition 6. Let (X, d) be a metric space with a metric d . A *discrete-time process* is a function $\phi : \mathbb{Z}_{\geq}^2 \times X \rightarrow X$ that satisfies

1. $\phi(t_0, t_0, x) = x \quad \forall t_0 \in \mathbb{Z}$ and $\forall x \in X$.
2. $\phi(t_2, t_0, x) = \phi(t_2, t_1, \phi(t_1, t_0, x)) \quad \forall t_0 \leq t_1 \leq t_2 \in \mathbb{Z}$ and $\forall x \in X$.
3. ϕ is continuous.

We now turn to formulating the driven kWTA difference equation (see Equation 1) as a discrete-time process. We first note that for a particular task, a set of possible inputs $P \cup \{0\}$ is defined. For completeness, this set covers the autonomous case by including the 0-vector. For each member of this set, we define a separate map $f^{(p)} : X \rightarrow X$ such that $f^{(p)}(x) = f(w \cdot x - \theta + \delta^{(p)})$. The set of maps $\{f^{(p)}\}$ with cardinality $|P| + 1$ defines a family of discrete-time autonomous semi-dynamical systems. These maps are chosen either randomly for the tasks $\text{RAND} \times 4$ and Parity-3 , or in a more structured fashion for the task Markov-85 . In either case, the kWTA discrete-time nonautonomous dynamics in Equation 1 can be rewritten in the form

$$x(t+1) = f^{(p_t)}(x(t)) = f(w \cdot x(t) - \theta + \delta^{(p_t)}(t)), \quad (15)$$

which generates a solution mapping

$$\phi(t_2, t_1, x(t_1)) = f^{(p_{t_2-1})} \circ \dots \circ f^{(p_{t_1})}(x(t_1)) \quad (16)$$

The solution mapping ϕ satisfies the three properties of a process. The first two properties are a product of the definition of function composition, and the continuity condition is proven exactly as in Proposition 3. Given the above, the family of discrete-time autonomous difference equations $\{f^{(p)}\}$ on the metric space (X, d) generates a process ϕ , and thus, it defines a particular kind of nonautonomous dynamical systems termed an *input-driven dynamical system*.

It is important to point out that an input-driven dynamical system is not defined for a particular input sequence, but for all input sequences drawn from its input set. This becomes more explicit if one considers the alternative *skew product* definition of a nonautonomous dynamical system, where the input is treated as a driving autonomous dynamical system [31,32]. We compare the two definitions of nonautonomous dynamical systems in Text S3. We now cover a few important concepts that will aid in defining the dynamic behavior of the model neural network.

Nonautonomous Attractors

Attractors in nonautonomous dynamical systems are defined on *nonautonomous sets*, relating strongly to the concepts of *invariance* and *entire solutions*.

Definition 7. Let $\phi : \mathbb{Z}_{\geq}^2 \times X \rightarrow X$ be a discrete-time input-driven dynamical system generated by the family of autonomous difference equations $\{f^{(p)}(x)\}$ on a metric space (X, d) . A subset $\mathcal{C} \subset \mathbb{Z} \times X$ is called a *nonautonomous set*, and for all $t \in \mathbb{Z}$, the set

$$\mathcal{C}_t = \{x \in X : (t, x) \in \mathcal{C}\}$$

is called the *t-fiber* of \mathcal{C} . \mathcal{C} is said to be *invariant* under ϕ if $\phi(t, t_0, \mathcal{C}_{t_0}) = \mathcal{C}_t$ for all $(t, t_0) \in \mathbb{Z}_{\geq}^2$. An *entire solution* of ϕ is an invariant set under ϕ whose t-fibers are the singleton sets $\{\xi(t)\}$ that are the images of the function $\xi : \mathbb{Z} \rightarrow X$ such that

$$\zeta(t_2) = \phi(t_2, t_1, \zeta(t_1)) = f^{(p_{t_2-1})} \circ \dots \circ f^{(p_{t_1})}(\zeta(t_1)) \quad \forall (t_2, t_1) \in \mathbb{Z}_{\geq}^2$$

$$\mathcal{V}_{\mathcal{A}}^{\omega;g} = \bigcup_{g(u_s)=\omega} \mathcal{V}_{\mathcal{A}}^{u_s}$$

An important property of invariant nonautonomous sets is that they consist exclusively of entire solutions (for a proof, see Lemma 2.15 in [31]). Nonautonomous attractors are nonautonomous sets. As such, they consist of entire solutions as well. There are several types of attractors of nonautonomous dynamical systems. Only of interest to our model neural network are forward attractors, so we drop the qualifier ‘forward’ and substitute it with ‘nonautonomous’.

Definition 8. Let $\phi : \mathbb{Z}_{\geq}^2 \times X \rightarrow X$ be a discrete-time input-driven dynamical system generated by the family of autonomous difference equations $\{f^{(p)}(x)\}$ on a metric space (X, d) . A nonautonomous set $\mathcal{A} \subset \mathbb{Z} \times X$ is a *nonautonomous attractor* of ϕ if the following conditions hold:

1. \mathcal{A} is invariant under ϕ .
2. \mathcal{A} is compact.
3. There exists a neighborhood \mathcal{N}_r of radius $r > 0$ such that $\lim_{t \rightarrow \infty} d(\phi(t, t_0, \mathcal{N}_r(\mathcal{A}_{t_0})), \mathcal{A}_t) = 0$ for all $t_0 \in \mathbb{Z}$

As in the autonomous dynamics of kWTA networks, all subsets of X are compact. The third condition assures that no subset of \mathcal{A} satisfies the invariance and compactness conditions. One may generalize the concept of a basin of attraction in an autonomous dynamical system to the nonautonomous case. This concept associates each nonautonomous attractor with the region of the state space that converges to that attractor:

Definition 9. Let $\phi : \mathbb{Z}_{\geq}^2 \times X \rightarrow X$ be a discrete-time input-driven dynamical system generated by the family of autonomous difference equations $\{f^{(p)}(x)\}$ on a metric space (X, d) . The *nonautonomous basin of attraction* of a nonautonomous attractor \mathcal{A} of ϕ is defined by

$$\mathcal{B}_{\mathcal{A}} = \{x \in X : \lim_{t \rightarrow \infty} d(\phi(t, t_0, x), \mathcal{A}_t) = 0 \text{ for all } t_0 \in \mathbb{Z}\}$$

Volumes of Representation

Spatiotemporal computations requires encoding different input sequences in the states of the neural network. The set of network states accessible from some initial conditions within a basin of attraction through perturbing the network with a particular input sequence $\langle p_1, \dots, p_s \rangle$ defines this sequence’s *volume of representation*.

Definition 10. Let $\phi : \mathbb{Z}_{\geq}^2 \times X \rightarrow X$ be a discrete-time input-driven dynamical system generated by the family of autonomous difference equations $\{f^{(p)}(x)\}$ on a metric space (X, d) . Given an input sequence $u_s = \langle p_1, \dots, p_s \rangle \in (P \cup \{0\})^s$ and a basin of attraction $\mathcal{B}_{\mathcal{A}}$, a subset

$$\begin{aligned} \mathcal{V}_{\mathcal{A}}^{u_s} &= \{x \in \mathcal{B}_{\mathcal{A}} : \exists x_0 \in \mathcal{B}_{\mathcal{A}} \text{ such that } x = \phi(s, 0, x_0) \\ &= f^{(p_s)} \circ \dots \circ f^{(p_1)}(x_0)\} \end{aligned}$$

is called the *volume of representation* of the input sequence u_s within the basin $\mathcal{B}_{\mathcal{A}}$. The sequence length $s \in \mathbb{Z}^+ \setminus \{0\}$ defines the *order* of this volume. The nonautonomous set $\mathcal{V}_{\mathcal{A}} \subset \mathbb{Z} \times \mathcal{B}_{\mathcal{A}}$ whose t-fibers are order-1 volumes of representation $\mathcal{V}_{\mathcal{A},t} = \mathcal{V}_{\mathcal{A}}^{p^{(t-1)}}$ is called the *perturbation set* within $\mathcal{B}_{\mathcal{A}}$. Also, given a function $g : P^s \rightarrow \Omega$ on input sequences such that $g(u_s) = \omega \in \Omega$, the set

is the volume of representation of ω given g .

It is straightforward to show that, within a basin of attraction, the volume of representation of some sequence $u_s = \langle p_1, \dots, p_s \rangle$ is a superset of the volume of a sequence $u_{s'+s} = \langle q_1, \dots, q_{s'}, p_1, \dots, p_s \rangle$ for all $u_{s'} = \langle q_1, \dots, q_{s'} \rangle \in (P \cup \{0\})^{s'}$, and that the volume of u_s is equivalent to the union of the volumes of $u_{s'+s}$ for all $u_{s'} \in (P \cup \{0\})^{s'}$. We term this property the *volumes’ inclusion property*.

The concept of ‘volumes of representation’ allows us to state the following theorem on the nature of attractors in discrete-time input-driven dynamical systems:

Theorem 11. Let $\phi : \mathbb{Z}_{\geq}^2 \times X \rightarrow X$ be a discrete-time input-driven dynamical system generated by the family of autonomous difference equations $\{f^{(p)}(x)\}$ on a metric space (X, d) , and let $\mathcal{B}_{\mathcal{A}}$ be a *compact* nonautonomous basin of attraction. The following holds:

1. The perturbation set $\mathcal{V}_{\mathcal{A}}$ is a superset of \mathcal{A} .
2. Within $\mathcal{B}_{\mathcal{A}}$, and for all $p \in P \cup \{0\}$, there exists one attractor $\mathcal{A}^{(p)} \subset \mathcal{V}_{\mathcal{A}}^p$ of the discrete-time autonomous semi-dynamical system generated by $f^{(p)}$.
3. $\mathcal{B}_{\mathcal{A}}$ is the basin of attraction of $\mathcal{A}^{(p)}$ for all $p \in P \cup \{0\}$.

Proof.

1. Since every attractor, whether autonomous or nonautonomous, is an invariant set, it is sufficient to prove that all invariant sets within a basin $\mathcal{B}_{\mathcal{A}}$ are a subset of its perturbation set $\mathcal{V}_{\mathcal{A}}$. Let’s consider an entire solution $\xi : \mathbb{Z} \rightarrow \mathcal{B}_{\mathcal{A}}$. For all $t_1 \in \mathbb{Z}$, it holds that $\xi(t_1 + 1) = \phi(t_1 + 1, t_1, \xi(t_1)) = f^{(p_{t_1})}(x(t_1)) \in \mathcal{V}^{p(t_1)}$. It follows by induction that $\xi(t_2 + 1) \in \mathcal{V}^{p(t_2)}$ for all $t_2 \geq t_1$. This translates to t-fibers of entire solutions being always a member of order-1 volumes of representation and that all entire solutions within a basin of attraction $\mathcal{B}_{\mathcal{A}}$ are subsets of the perturbation set $\mathcal{V}_{\mathcal{A}}$. Since invariant sets consist exclusively of entire solutions, it follows that all invariant sets are subsets of the perturbation set $\mathcal{V}_{\mathcal{A}}$, including the nonautonomous attractor \mathcal{A} .
2. Given some input p , we consider the discrete-time semi-dynamical system generated by $f^{(p)}$ on (X, d) with the solution mapping

$$\varphi(t, x) = \underbrace{f^{(p)} \circ f^{(p)} \circ \dots \circ f^{(p)}}_{t \text{ times}}(x)$$

From Definition 10 of volumes of representation, the order- s volume generated by p is the set $\mathcal{V}_{\mathcal{A}}^{p^s} = \varphi(s, \mathcal{B}_{\mathcal{A}}) \subseteq \mathcal{B}_{\mathcal{A}}$. Due to the compactness of $\mathcal{B}_{\mathcal{A}}$ and the continuity of $f^{(p)}$, $\mathcal{V}_{\mathcal{A}}^{p^s}$ is compact for all $s \in \mathbb{Z}^+$. Moreover, due to the volumes’ inclusion property, the family of compact volumes $(\mathcal{V}_{\mathcal{A}}^{p^s})_{s \in \mathbb{Z}^+}$ is nested with $\mathcal{V}_{\mathcal{A}}^0 = \mathcal{B}_{\mathcal{A}}$. As such, and according to Theorem 1.28 in [31], there exists a nonempty set

$$\mathcal{A}^{(p)} = \bigcap_{s \geq 0} \mathcal{V}_{\mathcal{A}}^{p^s}$$

that is both *compact* and *invariant* under $f^{(p)}$ and φ . It also follows

from the compactness of \mathcal{B}_A that $\mathcal{A}^{(p)}$ attracts \mathcal{B}_A , i.e. there exists $r > 0$ such that $\lim_{t \rightarrow \infty} d(\varphi(t, \mathcal{N}_r(\mathcal{B}_A)), \mathcal{A}^{(p)}) = 0$, and since $\mathcal{A}^{(p)}$ is a subset of \mathcal{B}_A , the neighborhood $\mathcal{N}_r(\mathcal{B}_A)$ is also a neighborhood of $\mathcal{A}^{(p)}$. Hence, the compact and invariant set $\mathcal{A}^{(p)}$ is an attractor of the discrete-time semi-dynamical system generated by $f^{(p)}$ and is a subset of \mathcal{V}_A^p .

- Since $\mathcal{A}^{(p)}$ attracts \mathcal{B}_A , it follows that the basin of attraction of $\mathcal{A}^{(p)}$ satisfies $\mathcal{B}_{\mathcal{A}^{(p)}} \supseteq \mathcal{B}_A$. Given a point $x \in \mathcal{B}_{\mathcal{A}^{(p)}} \setminus \mathcal{B}_A$, and since $\mathcal{A}^{(p)} \subset \mathcal{V}_A^p$, there exists $t \in \mathbb{Z}^+$ such that $\varphi(t, x) \in \mathcal{V}_A^p \subseteq \mathcal{B}_A$, which is a contradiction, since $x \notin \mathcal{B}_A$. Ergo, $\mathcal{B}_{\mathcal{A}^{(p)}} \setminus \mathcal{B}_A$ is an empty set, and \mathcal{B}_A is the basin of attraction of $\mathcal{A}^{(p)}$.

This theorem allows us to characterize the properties and relations between autonomous and nonautonomous attractors of kWTA networks, where all subsets of X are compact due to X 's finiteness and discreteness. Namely, it allows us, within some compact basin, to allocate the nonautonomous attractor's t-fibers as subsets of the t-fibers of the perturbation set, and it shows that the autonomous attractor of the input at time t is the t-fiber of the nonautonomous attractor.

Input-Insensitive Dynamics

It is possible for a process to behave locally or globally as an autonomous (semi-)dynamical system. That is equivalent, in the case of input-driven dynamical systems, to being input-insensitive.

Definition 12. Let $\phi : \mathbb{Z}_{\geq}^2 \times X \rightarrow X$ be a discrete-time input-driven dynamical system generated by the family of autonomous difference equations $\{f^{(p)}(x)\}$ on a metric space (X, d) . A state $x \in X$ is said to be *input-insensitive* if $f^{(p)}(x) = f^{(0)}(x)$ for all $p \in P$. An *input-insensitive basin* is a basin of attraction that consists entirely of input-insensitive states.

This definition implies that the volumes of representation of a particular order and the t-fibers of each nonautonomous set within this basin are equivalent, including the perturbation set and the nonautonomous attractor: they reduce to autonomous sets. The *input-insensitive attractor* becomes the autonomous attractor of each discrete-time semi-dynamical system generated by a difference equation $f^{(p)}$.

Supporting Information

Figure S1 Approximating volumes of representation using percentiles. (A) Percentile approximation of the order-1 volumes of representation of a SIP-RN. (B) Percentile approximation of the order-2 volumes of representation of a SIP-RN. Order-2 volumes are more exact approximations to the order-1 volumes according to the volumes' inclusion property. The correspondence is clarified by using similar color coding. (A,B) This approximation is done as follows. After transforming the network states to the principal components space, the coordinates of the first three principal components belonging to each volume of representation are first bootstrapped to 10000 samples, and the 5th and 95th percentiles are computed. Each volume is then approximated by an ellipsoid whose semi-axes extend to these percentiles and is centered at their average. This alternative approximation is less liberal than the one that uses means and standard deviations in that it extends the ellipsoids to assure including more true positives, but at the expense of including more false positives. One still sees, however, that the observations from the other approximation still hold, namely, that volumes

of representation are both redundant and separate from one another.

(TIF)

Figure S2 Volumes of representation of a nonlinear function over input sequences. Approximation of order-3 volumes of representation of the task Parity-3 binary input to a SIP-RN. By an appropriate union of these volumes, the volumes of representation of the outcome 0 (green) and 1 (orange) are identified. The approximation uses the mean and standard deviation of the coordinates. While the first three principal components are sufficient for showing distinct order-3 volumes of representation, more dimensions are necessary to illustrate separate volumes of the outcome of the nonlinear function. The separability of the function's outcomes explains the ability of optimal linear classifiers to successfully perform the nonlinear task. (TIF)

Figure S3 Average classification performance using the Hamming distance of the network states from the vertexes of autonomous attractors. 100 networks are trained by STDP and IP simultaneously on (A) the memory task RAND x 4, (B) the prediction task Markov-85, and (C) the nonlinear task Parity-3. Given the input set P , and the family of discrete-time autonomous semi-dynamical systems generating these networks $\{f^{(p)}(x)\}$, the network states comprising the autonomous attractor (the attractor's vertexes) are identified as follows. First, initial conditions are selected within the input-sensitive basin of attraction. Second, the input is clamped to one member of P . Third, the solution of $f^{(p)}(x)$ is generated for a sufficient number of time steps, so that the dynamics, following a transient period, converges to the attractor. Training and testing optimal linear classifiers is carried through as before. The training and testing data is, however, the Hamming distance between the network states and the vertexes of the attractors. Error bars indicate standard error of the mean. The red line marks chance level. The x-axis shows the input time-lag. Negative time-lags indicate the past, and positive ones, the future. (TIF)

Figure S4 Average classification performance of networks combining the weights of SP-RNs and thresholds of IP-RNs. 100 networks are trained by STDP and IP simultaneously (orange), IP alone (blue), or trained by STDP alone followed by injecting the thresholds resulting from IP at the end of the plasticity phase (green) on (A) the memory task RAND x 4, (B) the prediction task Markov-85, and (C) the nonlinear task Parity-3. The combined networks (green) lack the contribution of the interaction between synaptic and intrinsic plasticity during the plasticity phase. This results in their performance being inferior to the networks where synaptic and intrinsic plasticity interact. Error bars indicate standard error of the mean. The red line marks chance level. The x-axis shows the input time-lag. Negative time-lags indicate the past, and positive ones, the future. (TIF)

Text S1 Comparing nonplastic networks.
(PDF)

Text S2 Long-term behavior of learning.
(PDF)

Text S3 Definitions of nonautonomous dynamical systems.
(PDF)

Acknowledgments

Discussions with Johannes Schumacher are gratefully acknowledged.

References

- Cooper LN, Intrator N, Blais BS, Shouval HZ (2004) Theory of cortical plasticity. World Scientific.
- Broome BM, Jayaraman V, Laurent G (2006) Encoding and decoding of overlapping odor sequences. *Neuron* 51: 467–482.
- Buonomano DV, Maass W (2009) State-dependent computations: spatiotemporal processing in cortical networks. *Nat Rev Neurosci* 10: 113–125.
- Nikolić D, Häusser S, Singer W, Maass W (2009) Distributed fading memory for stimulus properties in the primary visual cortex. *PLoS Biol* 7: e1000260.
- Abbott LF, Nelson SB (2000) Synaptic plasticity: taming the beast. *Nat Neurosci* 3: 1178–1183.
- Zhang W, Linden DJ (2003) The other side of the engram: experience-driven changes in neuronal intrinsic excitability. *Nat Rev Neurosci* 4: 885–900.
- Turrigiano GG, Nelson SB (2004) Homeostatic plasticity in the developing nervous system. *Nat Rev Neurosci* 5: 97–107.
- Rumelhart DE, McClelland JL (1986) Parallel distributed processing: explorations in the microstructure of cognition. 2 volumes. Cambridge, MA: MIT Press.
- Izhikevich EM (2006) Polychronization: computation with spikes. *Neural Comput* 18: 245–282.
- Shadlen MN, Newsome WT (1994) Noise, neural codes and cortical organization. *Curr Opin Neurobiol* 4: 569–579.
- Gerstner W, Kistler WM (2002) Spiking neuron models: Single neurons, populations, plasticity. Cambridge University Press.
- Stein RB, Gossen ER, Jones KE (2005) Neuronal variability: noise or part of the signal? *Nat Rev Neurosci* 6: 389–397.
- Rolls ET, Deco G (2010) The noisy brain: stochastic dynamics as a principle of brain function. Volume 28. New York: Oxford University Press.
- Bienenstock EL, Cooper LN, Munro PW (1982) Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J Neurosci* 2: 32–48.
- Triesch J (2007) Synergies between intrinsic and synaptic plasticity mechanisms. *Neural Comput* 19: 885–909.
- Toyoizumi T, Pfister JP, Aihara K, Gerstner W (2005) Generalized bienenstock-cooper-munro rule for spiking neurons that maximizes information transmission. *Proc Natl Acad Sci U S A* 102: 5239–5244.
- Savin C, Joshi P, Triesch J (2010) Independent component analysis in spiking neurons. *PLoS Comput Biol* 6: e1000757.
- Clopath C, Büsing L, Vasilaki E, Gerstner W (2010) Connectivity recodes coding: a model of voltage-based STDP with homeostasis. *Nat Neurosci* 13: 344–352.
- Buesing L, Maass W (2010) A spiking neuron as information bottleneck. *Neural Comput* 22: 1961–1992.
- Douglas RJ, Martin K (1991) A functional microcircuit for cat visual cortex. *J Physiol* 440: 735–769.
- Douglas RJ, Koch C, Mahowald M, Martin K, Suarez HH (1995) Recurrent excitation in neocortical circuits. *Science* 269: 981–985.
- Douglas RJ, Martin KA (2004) Neuronal circuits of the neocortex. *Annu Rev Neurosci* 27: 419–451.
- Weiler N, Wood L, Yu J, Solla SA, Shepherd GM (2008) Top-down laminar organization of the excitatory network in motor cortex. *Nat Neurosci* 11: 360–366.
- DeAngelis GC, Ohzawa I, Freeman R (1993) Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex. ii. linearity of temporal and spatial summation. *J Neurophysiol* 69: 1118–1135.
- Christopher deCharms R, Blake DT, Merzenich MM (1998) Optimizing sound features for cortical neurons. *Science* 280: 1439–1444.
- Mauk MD, Buonomano DV (2004) The neural basis of temporal processing. *Annu Rev Neurosci* 27: 307–340.
- Lazar A, Pipa G, Triesch J (2007) Fading memory and time series prediction in recurrent networks with different forms of plasticity. *Neural Netw* 20: 312–322.
- Lazar A, Pipa G, Triesch J (2009) Sorn: a self-organizing recurrent neural network. *Front Comput Neurosci* 3: 23.
- Jaeger H (2001) The “echo state” approach to analysing and training recurrent neural networks. Techn. rep. GMD 148, Bremen: German National Research Center for Information Technology.
- Maass W, Natschläger T, Markram H (2002) Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Comput* 14: 2531–2560.

Author Contributions

Conceived and designed the experiments: HT GP. Performed the experiments: HT. Analyzed the data: HT. Contributed reagents/materials/analysis tools: HT. Wrote the paper: HT GP.

- Kloeden PE, Rasmussen M (2011) Nonautonomous dynamical systems. AMS Bookstore.
- Kloeden PE, Pötzsche C, Rasmussen M (2013) Discrete-time nonautonomous dynamical systems. In: Johnson R, Pera M, editors. *Stability and Bifurcation Theory for Non-Autonomous Differential Equations*. Springer. pp. 35–102.
- Cover TM, Thomas JA (2006) Elements of information theory. 2nd edition. John Wiley & Sons.
- Strogatz SH (2001) Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering (studies in nonlinearity). Westview Press.
- Brunel N (2000) Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J Comput Neurosci* 8: 183–208.
- Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci U S A* 79: 2554–2558.
- Pasemann F (1995) Characterization of periodic attractors in neural ring networks. *Neural Netw* 8: 421–429.
- Rabinovich M, Volkovskii A, Lecaña P, Huerta R, Abarbanel H, et al. (2001) Dynamical encoding by networks of competing neuron groups: winnerless competition. *Phys Rev Lett* 87: 068102.
- Rabinovich MI, Huerta R, Varona P, Afraimovich VS (2008) Transient cognitive dynamics, metastability, and decision making. *PLoS Comput Biol* 4: e1000072.
- Timme M, Wolf F, Geisel T (2002) Prevalence of unstable attractors in networks of pulse-coupled oscillators. *Phys Rev Lett* 89: 154105.
- Skarda CA, Freeman WJ (1987) How brains make chaos in order to make sense of the world. *Behav Brain Sci* 10: 161–195.
- Gros C (2009) Cognitive computation with autonomously active neural networks: an emerging field. *Cognit Comput* 1: 77–90.
- Marković D, Gros C (2012) Intrinsic adaptation in autonomous recurrent neural networks. *Neural Comput* 24: 523–540.
- Negrello M, Pasemann F (2008) Attractor landscapes and active tracking: The neurodynamics of embodied action. *Adapt Behav* 16: 196–216.
- Pascanu R, Jaeger H (2011) A neurodynamical model for working memory. *Neural Netw* 24: 199–207.
- Manjunath G, Jaeger H (2013) Echo state property linked to an input: Exploring a fundamental characteristic of recurrent neural networks. *Neural Comput* 25: 671–696.
- Panzeri S, Schultz SR, Treves A, Rolls ET (1999) Correlations and the encoding of information in the nervous system. *Proc R Soc Lond B Biol Sci* 266: 1001–1012.
- Narayanan NS, Kimchi EY, Laubach M (2005) Redundancy and synergy of neuronal ensembles in motor cortex. *J neurosci* 25: 4207–4216.
- Puchalla JL, Schneidman E, Harris RA, Berry MJ (2005) Redundancy in the population code of the retina. *Neuron* 46: 493–504.
- Chechik G, Anderson MJ, Bar-Yosef O, Young ED, Tishby N, et al. (2006) Reduction of information redundancy in the ascending auditory pathway. *Neuron* 51: 359–368.
- Tkačik G, Prentice JS, Balasubramanian V, Schneidman E (2010) Optimal population coding by noisy spiking neurons. *Proc Natl Acad Sci U S A* 107: 14419–14424.
- Kenet T, Bibitchkov D, Tsodyks M, Grinvald A, Arieli A (2003) Spontaneously emerging cortical representations of visual attributes. *Nature* 425: 954–956.
- Fiser J, Chiu C, Weliky M (2004) Small modulation of ongoing cortical dynamics by sensory input during natural vision. *Nature* 431: 573–578.
- Pfister JP, Tass PA (2010) STDP in oscillatory recurrent networks: theoretical conditions for desynchronization and applications to deep brain stimulation. *Front Comput Neurosci* 4: pii: 22.
- Brea J, Senn W, Pfister JP (2013) Matching recall and storage in sequence learning with spiking neural networks. *J Neurosci* 33: 9565–9575.
- Markram H, Lübke J, Frotscher M, Sakmann B (1997) Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science* 275: 213–215.
- Song S, Miller KD, Abbott LF (2000) Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nat Neurosci* 3: 919–926.
- Kraskov A, Stögbauer H, Grassberger P (2004) Estimating mutual information. *Phys Rev E Stat Nonlin Soft Matter Phys* 69: 066138.