



Shaping dynamical neural computations using spatiotemporal constraints

Jason Z. Kim^{a,*}, Bart Larsen^b, Linden Parkes^{c,**}

^a Department of Physics, Cornell University, Ithaca, NY, 14853, USA

^b Department of Pediatrics, Masonic Institute for the Developing Brain, University of Minnesota, USA

^c Department of Psychiatry, Rutgers University, Piscataway, NJ, 08854, USA



ABSTRACT

Dynamics play a critical role in computation. The principled evolution of states over time enables both biological and artificial networks to represent and integrate information to make decisions. In the past few decades, significant multidisciplinary progress has been made in bridging the gap between how we understand biological *versus* artificial computation, including how insights gained from one can translate to the other. Research has revealed that neurobiology is a key determinant of brain network architecture, which gives rise to spatiotemporally constrained patterns of activity that underlie computation. Here, we discuss how neural systems use dynamics for computation, and claim that the biological constraints that shape brain networks may be leveraged to improve the implementation of artificial neural networks. To formalize this discussion, we consider a natural artificial analog of the brain that has been used extensively to model neural computation: the recurrent neural network (RNN). In both the brain and the RNN, we emphasize the common computational substrate atop which dynamics occur—the connectivity between neurons—and we explore the unique computational advantages offered by biophysical constraints such as resource efficiency, spatial embedding, and neurodevelopment.

1. Introduction

Dynamics have long underpinned computation. From the cycles of central pattern generators that support locomotion [1] to the networks of large-scale brain dynamics thought to regulate decision-making [2], it is clear that biological systems make ample use of their time-evolution to respond to their environment. Harnessing this dynamical computation, artificial recurrent neural networks (RNNs) have been trained to successfully perform the same computational tasks as humans [3,4]. However, while inspired by the brain, training of RNNs is typically carried out in an unconstrained manner, leading to solutions that lack biophysical realism. Additionally, decades of neuroscience research has demonstrated the importance of biological constraints for achieving the brain's unique structure and capabilities [5–8]. Here, we draw on literature from dynamical systems and neuroscience to discuss (i) how RNNs leverage dynamics to compute, and (ii) how biophysical constraints may shape this computation by guiding the formation of network structure. At the intersection of these goals exists an opportunity to study how biologically-constrained RNNs may yield more powerful and more interpretable computational models.

To understand how biological neurons compute, there has been a long and rich history of modeling and interpreting neurobiological systems to leverage their computational capabilities. These quantitative

models fall under the category of *dynamical systems*, whose evolution in time is determined by mathematical functions. At the scale of a single neuron, detailed circuit models of the ion channels that mediate membrane voltage provide quantitative understanding of signal propagation and computation in dendrites [9,10]. At the scale of neural populations, mean-field models of excitatory and inhibitory neurons have enabled the study of neural circuits for biological sensing, imitation, and attention [11–13]. At the whole-brain level, both linear [14–17] and non-linear [18,19] dynamical models have been used to simulate large-scale activity patterns, and have examined how those patterns spread across the brain's white matter tracts. Across this broad range of systems, scales, and models, there exist a diversity of ways in which dynamics can be used for computation, and a crucial dependence of these dynamics on biophysical parameters.

More recently, with technological advances in deep learning, the study of neural computation has adopted a more functional direction that moves away from biological realism. That is, rather than seeking a direct biophysical model [9,11,20,21], RNNs posit a general dynamical system that is Turing complete [22], with parameters that are trained to solve computational tasks. We focus on RNNs because they have been used extensively to understand how general brain-like systems leverage dynamics to perform computation. Examples of this use include time-series prediction [23], source-separation [24], decision-making

* Corresponding author.

** Corresponding author.

E-mail addresses: jk2557@cornell.edu (J.Z. Kim), linden.parkes@rutgers.edu (L. Parkes).

[3], odor classification [25], as well as concurrent performance of multiple cognitive tasks [26]. However, these insights and models often fail to translate into the real computational substrate of the brain—the neural architecture—because RNNs are trained without regard for biophysical constraints.

To merge biological realism with computational dynamics, we must first understand the physical embedding and constraints of the brain. In contrast to artificial RNNs, the brain is embedded within a circumscribed physical space [27], and its inter-connectivity is subject to limited metabolic resources [28]. This discrepancy makes it challenging to translate insights relating the structure, dynamics, and computation of biological brains to artificial RNNs. Neuroscience has studied these resource constraints for more than a century [29], suggesting that the brain is pressured to make efficient use of space, material, and time. That is, the brain must learn to communicate efficiently (time) while leveraging limited physical (space) as well as metabolic and cellular (material) resources. Critically, many of the brain's topological features of connectivity and communication are thought to emerge as a consequence of navigating these pressures [5–8,28]. These findings suggest that the brain's resource constraints play a critical role in shaping its dynamic repertoire and computational capacity.

Here, our goal is to lay out promising new directions for improving the computational power and interpretability of RNN models of the brain. We posit that this goal will be achieved by placing biological constraints on RNNs that shape their structure and activity in systematic ways, which will in turn produce computationally improved dynamics. We focus on two aspects of RNN computational dynamics: the diversity of information that is represented by the neurons (expressivity), and the manipulation of low-dimensional internal representations (latent-spaces). In each section, we examine how biology shapes brain networks—with particular emphasis on the spatially-patterned macro-scale organizing principles of the cortex—and discuss how these constraints may be ported to RNNs to improve performance with interpretable structure and dynamics. Overall, we discuss how insights from biological and artificial computation can enrich each other towards a new generation of biophysically realistic RNNs.

2. The RNN model

To mathematically model the time-evolution of neural systems, we turn to dynamical systems which posit that the next state of a neural system can be written as a function of the current state and an input as

$$\mathbf{r}_{t+1} = f(\mathbf{r}_t, \mathbf{u}_t). \quad (1)$$

Here, $\mathbf{r}_t \in \mathcal{R}^n$ is a vector of n neural activity states, $\mathbf{u}_t \in \mathcal{R}^k$ is a vector of k inputs, and f is a function. As an example, let us consider a simple leaky integrator model with a single neuron which evolves according to

$$\mathbf{r}_{t+1} = a\mathbf{r}_t + b\mathbf{u}_t, \quad (2)$$

where $0 \leq a < 1$ and b are real numbers (Fig. 1A). As time evolves forward, the neuron state integrates the input $b\mathbf{u}_t$, and the accumulated history of inputs decays at a rate set by a .

While Eq. (1) is written with t advancing in integer steps—thereby called a discrete-time dynamical system—many physical neural models evolve forward continuously in time as,

$$\frac{d}{dt}\mathbf{r}(t) = \hat{f}(\mathbf{r}(t), \mathbf{u}(t)). \quad (3)$$

We can approximate these continuous dynamics as discrete by evolving Eq. (3) in time using steps of Δt as,

$$\mathbf{r}_{t+\Delta t} = f(\mathbf{r}_t, \mathbf{u}_t) = \mathbf{r}_t + \int_t^{t+\Delta t} \hat{f}(\mathbf{r}(\tau), \mathbf{u}(\tau)) d\tau. \quad (4)$$

For example, the continuous-time version of the leaky integrator neuron is given by

$$\frac{d}{dt}r(t) = \hat{a}r(t) + \hat{b}u(t), \quad (5)$$

for $\hat{a} < 0$ (Fig. 1A). In this case, the parameters of these two models can be interchanged through the transformation $a = e^{\hat{a}\Delta t}$, $b = (a - 1)\hat{b}/\hat{a}$, but fundamental differences exist between continuous- and discrete-time systems [30]. Regardless of the system type, neural models make

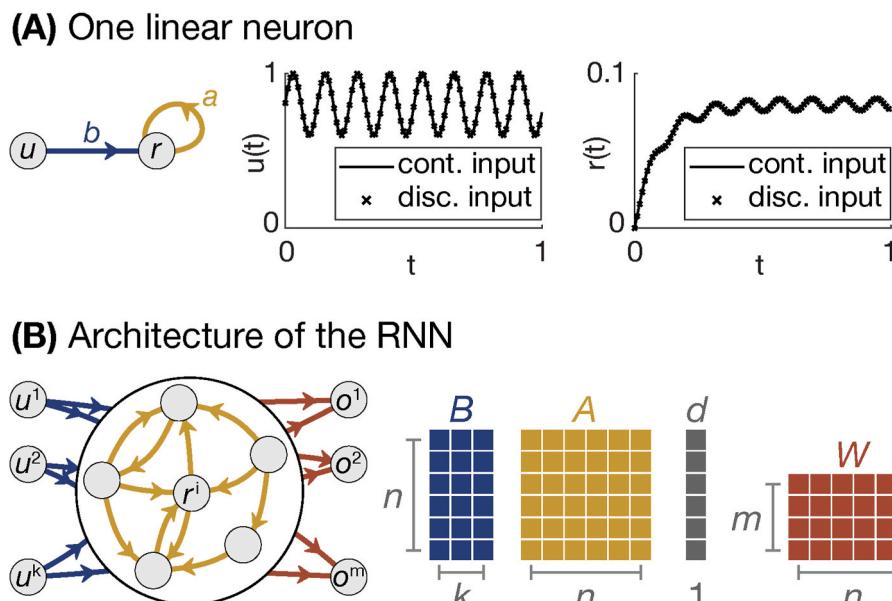


Fig. 1. One-dimensional dynamics and RNN architecture. A, Schematic of a single neuron model of a leaky integrator with an input connection in blue, and a recurrent connection in gold (left), alongside examples of both a continuous and discretized input into the neuron (center), and the state of the input-driven neuron from the continuous (Eq. (5)) and discretized (Eq. (2)) dynamics (right). B, Schematic of an RNN with k inputs and m outputs, with input connections and matrix in blue, recurrent connections and matrix in gold, and output connections and matrix in maroon. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

tradeoffs between complexity in the level of detail and tractability.

The RNN is a model that attempts to capture the biophysical quantity of the *interactions* between neurons through the connectivity matrix A . In tandem, the RNN simplifies the precise functional form of that interaction through the activation function f . In its most basic form, an RNN is a subset of dynamical systems (Eq. (3)) that evolves in time as

$$\begin{aligned} \mathbf{r}_{t+1} &= f(A\mathbf{r}_t + B\mathbf{u}_t + \mathbf{d}), \\ \mathbf{o}_t &= g(\mathbf{r}_t). \end{aligned} \quad (6)$$

where $A \in \mathbb{R}^{n \times n}$ is the connectivity matrix between neurons, $B \in \mathbb{R}^{n \times k}$ is a matrix that linearly maps the inputs to the neurons, $\mathbf{d} \in \mathbb{R}^n$ is a vector of bias terms, and f is an activation function (Fig. 1B). Rather than having f be a complex and biophysically realistic function, it is often approximated as a simple nonlinear function such as a sigmoid. The output of the RNN, \mathbf{o}_t , is usually taken to be some function g of the RNN state, and is often a linear output $\mathbf{o}_t = W\mathbf{r}_t$. Typically, A , B , \mathbf{d} , and W are treated as learnable parameters, some or all of which can be trained using a wide variety of methods [4,31,32]. An important class of RNNs is in the limit where only the output weights W are trained, called *reservoir computing* [33], which is now gaining traction for biophysical modeling by using connectomes as the connectivity matrix [34].

We broadly organize the following sections around two computational perspectives and how they manifest in and are improved by biology. In both, the standard task is to input data into the RNN, and train some or all of its parameters to output a particular function of the input. The first perspective is to treat the RNN as an input-output machine that generates a complex representation of the data. We center the discussion around the space of functions the RNN can represent, or its *expressivity*. The framework that relies most heavily on this perspective is reservoir computing because only the output weights are trained, such that the allowed computations are completely determined by its expressivity (Section III). The second perspective is to treat the RNN as a computing machine where the internal dynamics not only represent the data, but repeatedly transform it in a principled manner according to some algorithm. We center this discussion around the RNN's low-dimensional dynamics, or its *latent space attractors*. The framework that relies most heavily on this perspective is the standard RNN with all trainable parameters, because its ability to modify its internal connectivity allows it to adapt its dynamics (Section IV).

Importantly, these reliances are far from absolute. Recent work demonstrates the ability of reservoir computers to easily construct incredibly sophisticated latent-space computations and attractors through a very simple feedback update [35,36]. Conversely, the latent space attractors of RNNs with fully trainable parameters must necessarily exist within the space of expressible functions. Hence, in equal measure to outlining unique biological and mathematical directions of future work for each perspective, we seek to outline such directions at the intersection of these perspectives.

3. Tuning expressivity through regularized activity and neuromodulation

When solving any computational problem, the expressivity of the language used is of crucial importance. The more expressive a language is, the greater the set of computations, formulae, and theorems comprise that language [37,38]. For example, a programming language that supports conditional statements can represent many more programs than an equivalent language that lacks conditional statements.

In the same way, neural networks can be viewed from the lens of expressivity. Generally, what we mean by expressivity is the space of all possible functions that can be learned by a neural network. Intuitively, architectural features such as the number of neurons and the topology of the layers determine the diversity of the space of learnable functions. Previous work has demonstrated that shallow multi-layer perceptrons (MLPs) are universal function approximators [39,40], and that RNNs are

universal dynamics approximators [41]. However, even if a particular function or dynamical pattern is theoretically achievable, artificial neural networks—just like biological neural networks—must be trained from an initial condition. Hence, the study of expressivity extends beyond theoretical guarantees, and has been shown to rely heavily upon architectural features such as depth [42], neuron activation [43], and connectivity [44], which are crucial for explaining and engineering the success of modern-day neural networks.

In this section, we study three consequences of biological processes for expressivity. First, we introduce RNN expressivity as a richness of time history information about the inputs, and tie this richness to spatially-patterned temporal receptive fields in the brain that underpin information integration. Second, we study constraints on expressivity induced by resource constraints on neural activity as a putative learning mechanism. Finally, we explore the potential for neural networks to modulate their expressivity at short time-scales through neuromodulation. Together, the processes of the brain offer enticing and novel paradigms for training and constructing more expressive RNNs under biological constraints.

3.1. Expressivity as variable time-lagged integration of information

Expressivity of RNNs is intricately tied to the concept of *stability*: how quickly a perturbation to the RNN state decays [45]. If a perturbation decays very quickly, then the information contained in the perturbation cannot be used by the RNN for extended information processing. On the other hand, if the perturbation grows uncontrollably, then the temporal information in the inputs quickly becomes too complex to be represented by the finite number of neurons. As a result, an optimal amount of controlled stability should maximally preserve temporal information without saturating the RNN's capacity.

We can quantify this intuition through a simple recursive substitution of Eq. (6),

$$\begin{aligned} \mathbf{r}_{t+1} &= f(A\mathbf{r}_t + B\mathbf{u}_t + \mathbf{d}) \\ &= f(A(f(A\mathbf{r}_{t-1} + B\mathbf{u}_{t-1} + \mathbf{d}) + B\mathbf{u}_t + \mathbf{d}) \\ &\quad = \dots \\ &= h(\mathbf{r}_0, \mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_t). \end{aligned} \quad (7)$$

Hence, in a noiseless system, the RNN state \mathbf{r}_{t+1} can be written as an explicit function h of the initial state, \mathbf{r}_0 , and the full time history of the inputs, \mathbf{u}_t , is mediated by the recursive application of A , B , \mathbf{d} , and the activation function f . The state of all neurons \mathbf{r}_{t+1} generates a *basis* for a subspace of the delay-embedded space of inputs \mathbf{u}_t (Fig. 2A), which means that the neuron states at time $t+1$ hold information about the time history of the inputs mediated by A , B , and \mathbf{d} [36]. Thus, when computing an *output* using the neural states, we are implicitly computing an output using a basis of time-lagged input terms, where the connectivity defines the basis vectors, and therefore the reconstructable subspace (Fig. 2A). The more expressive this time-lagged basis, the greater the diversity of output functions which can be computed.

This expressivity is intimately tied to the connectivity matrix [46], and has been studied through many different lenses such as computation at the edge of chaos [44,47,48], criticality, and avalanches [49,50]. The RNN's stability is set by the specific activation function f and the connectivity matrix A . To gain intuition for this dependence, we consider a linear 2 neuron system driven by one input. When there is no connectivity between the neurons, they store no time history of the inputs, and their state at time $t=3$ is purely a function of the input at time $t=2$ (Fig. 2B). When we add weak connections between the neurons, they begin to store some information about the input at the previous time point $t=1$ (Fig. 2C). When we strengthen these connections, the neurons begin to store information from further back in time at $t=0$ (Fig. 2D).

To further develop this intuition for larger systems and a specific task, we consider a 50 neuron system whose connectivity is randomly initialized, and whose output is trained to recall an impulse from 30 time

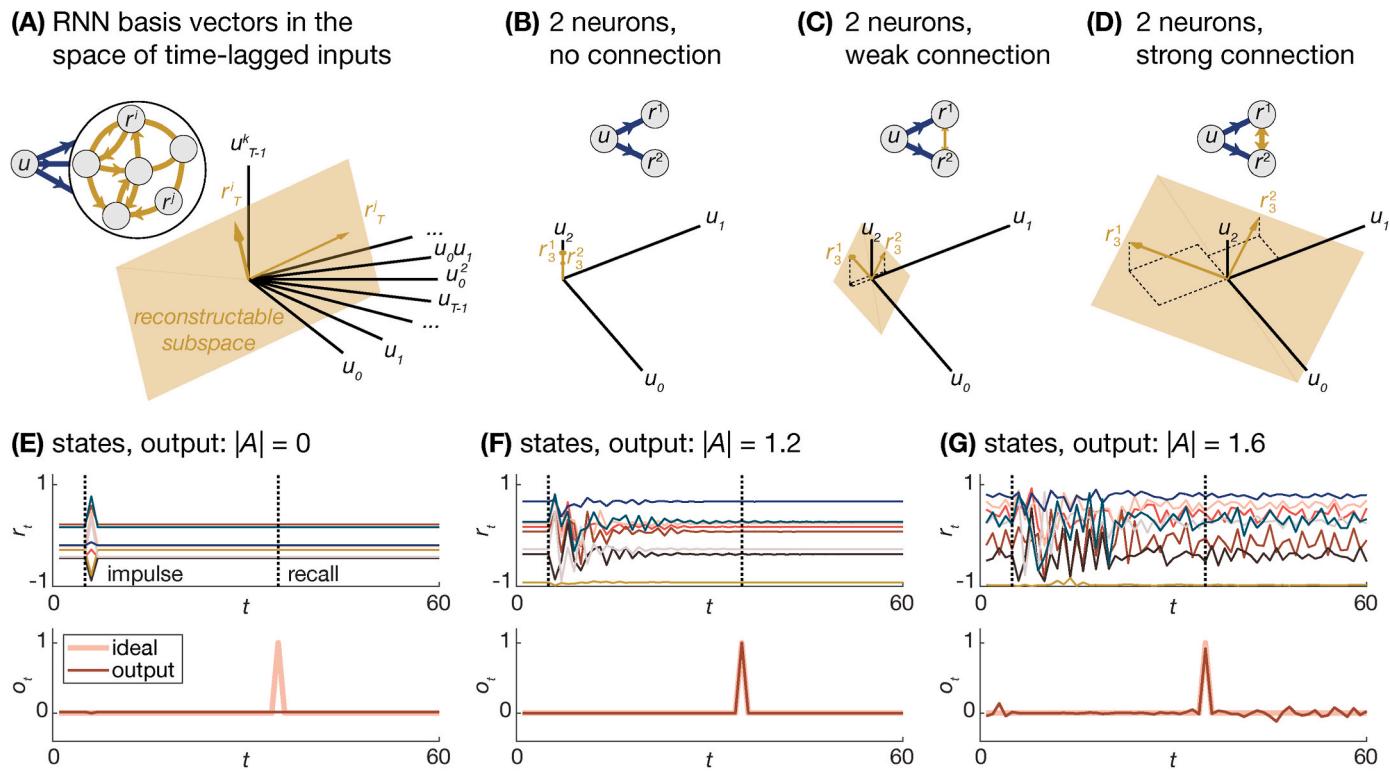


Fig. 2. Expressivity and the representational basis formed by RNNs. A Schematic of an RNN with a single input, the basis vectors formed by neurons i and j , and the reconstructable subspace spanned by those vectors. B Schematic and reconstructable subspaces for a simple 2 neuron system with no recurrent connections, C weak recurrent connections, and D strong recurrent connections. E Example trajectories from a subset of 8 neurons of a 50 neuron system after receiving an impulse (top), alongside the ideal and true output of the RNN (bottom) after training a linear readout, $o_t = Wr_t$, to reproduce a time-lagged version of the impulse with $|A| = 0$, F $|A| = 1.2$, and G $|A| = 1.6$.

steps in the past. By recall, we mean that the RNN output should be 0 at all time points except at 30 time steps after the impulse, thereby remembering the impulse from 30 time steps ago. At the trivial limit of an RNN with no connectivity where $|A| = 0$, the recursion of Eq. (7) yields $r_{t+1} = f(Bu_t + d)$, and we see that there is no time history of the input present in the RNN state. As a result, the RNN is unable to recall the input at a later point in time (Fig. 2E) [51]. As we increase the strength of connectivity, the RNN state stores more information about longer time lags of the input, $u_{t-\tau}$, and is thus able to more accurately recall the input later in time (Fig. 2F). As the connectivity strength continues to further increase, the RNN state holds increasingly more time lags of the input until it saturates such that the number of neurons is smaller than the dimension of the space of time-lagged input functions, thereby forming an incomplete basis for that space (Fig. 2G). This incomplete basis is a crucial reason why it is beneficial for the RNN's memory of an input to fade with time to achieve optimal stability [52]. Hence, this framework of expressivity provides a rich and quantitative perspective for studying the computational impact of imposing biophysical costs to the RNN connectivity.

This link between storing long time histories and expressivity is displayed prominently and spatially in the brain. Specifically, there exists a tight coupling between longer periods of temporal integration and higher-order computation, and this relationship varies systematically across the cortex [53]. At the macro-scale, cortical brain regions are thought to follow a dominant axis of variation that encodes a global processing hierarchy [54]. This gradient of brain organization is broadly referred to as the sensorimotor-association (S-A) axis [55,56]. The S-A axis spans from primary cortices supporting sensation and movement at the bottom, to multimodal cortices supporting multisensory processing and integration in the middle, to transmodal association cortices supporting higher-order cognition at the top. The S-A axis is observed across multiple diverse features of brain structure and function [55] and is

conserved across species [56,57], demonstrating its evolutionary roots. Notably, as regions traverse up the S-A axis they undergo a progressive lengthening of their temporal receptive windows [53,58]. Specifically, regions' intrinsic functional timescales vary over the S-A axis [59,60] with regions at the top showing slower fluctuations reflecting longer temporal receptive windows. In turn, these longer windows are thought to enable greater accumulation and integration of information over time, facilitating higher-order cognition [53]. Conversely, regions at the bottom of the S-A axis show relatively fast dynamics, which is thought to underpin rapid integration of recent sensory information [53].

This spatial patterning of receptive windows suggests that the brain—unlike naively constructed RNNs—distributes its computational expressivity systematically across the cortex, and research suggests that this may be critical for functional integration [53]. The above data suggest that RNNs too may benefit from spatially varying periods of temporal integration. Specifically, the stability of RNNs is typically only considered globally across the entire system, as the connectivity of many RNNs are initialized randomly. If the RNN is linear (i.e., if the activation function f is the identity matrix), then any ensuing dynamic instabilities are localized to linear subspaces, or *modes*, of neural activity [61]. However, if the RNN is nonlinear (e.g., $f = \tanh$), then instabilities bleed into other modes, making it difficult for the RNN to form a clean segregation of time-scales. Hence, varying periods of temporal integration could be achieved by specifying spatially varying penalties on neuronal timescales into the RNN cost functions. Such penalties may give rise to spatially segregated modules responsible for processing inputs at different time-scales.

3.2. Learning by suppressing activity

While expressivity hinges on a careful balance of stable dynamics, biological neural networks are constrained by energy; more active

neurons require more metabolic energy, which is a limited resource. As such, while artificial networks can maximize their expressivity through unconstrained backpropagation, the brain's capacity to learn is restricted by resource constraints; also, the extent to which the brain performs backpropagation remains unclear [62], which has motivated the machine learning community to consider more biologically-inspired optimization approaches. Hence, penalizing activity in RNNs should intuitively penalize computational capability through reduced expressivity. However, recent work has instead demonstrated important computational benefits of minimizing energy usage during training [63]. For example, Ali et al. [63] trained an RNN to predict sequences of handwritten digits, and examined how different optimization functions impacted model architecture and behavior. Specifically, Ali et al. [63] did not train their RNN to minimize prediction error through backpropagation. Instead, they trained their model to minimize absolute levels of neural activity prior to passing that activity through neurons' activation functions (here, ReLU). Such *preactivation minimization* is akin to selectively minimizing neurons' presynaptic inputs in biological networks. Critically, executing this cost function required no information about task performance, and instead simply limited the RNN's resources in a biologically plausible way.

Alongside good task performance, Ali et al. [63] observed dynamics in their RNN indicative of predictive coding. Predictive coding describes the hypothesis that the brain stores and updates expectations about its environment which it compares with incoming sensory evidence for those expectations [64]. Specifically, Ali et al. [63] observed activity patterns in their RNN suggestive of (i) selective self-inhibition in neurons receiving visual stimuli and (ii) prediction of future inputs in neurons not receiving visual stimuli. These results accord with hierarchically-organized predictive coding coupled to the S-A axis, wherein association cortices store predictions and via their distributed connectivity modulate activity in sensorimotor cortices [65–67]. Taken together, the findings of Ali et al. [63] indicate that while limiting the neurons' activation might intuitively reduce their expressivity—thereby limiting their computational capability in RNNs—it can also serve as a unique mechanism for distributed learning that is more biophysically realistic than backpropagation.

3.3. Dynamically tuning expressivity via neuromodulation

The preceding sections discussed expressivity as an emergent property of a trained network that can be modified by placing certain constraints on RNN training. This static expressivity has been shown to be effective at performing a wide range of tasks, and is the basis of the success of *reservoir computing* [33]. Unlike in an RNN, the internal connectivity of the reservoir computer (RC) is not trained. Instead, only the output is trained, typically as a weighted sum of RC states [31]. Hence, RCs rely completely on the preexisting expressivity of their internal dynamics to generate a sufficiently expressive basis representation of their inputs. Because RCs can be trained without knowledge or modification of the internal system, a wide variety of physical systems have been explored as efficient RCs [68], including photonics [69], electrical circuits [70], hydrodynamic systems [24], and the brain [71]. However, expressivity in biological networks is not static, even in the presence of fixed weights. Instead it can be modulated dynamically over short time-scales and—similar to regions' temporal receptive windows—this too is spatially patterned.

Previously, we discussed how the S-A axis tracks functional specialization and integration across the cortical mantle; cortical brain systems located at the bottom of the S-A axis are responsible for processing sensory/motor information while systems at the top of the S-A axis are involved in processing higher-order cognition [54], and the brain's connectivity allows for the hierarchical flow of information across these systems [72–74]. However, the functional roles of these different brain systems are not static. Instead the brain utilizes a complex array of neuromodulatory systems to actively reconfigure the

brain's dynamic repertoire [75]. In turn, this neuromodulation endows a relatively static network architecture (i.e., structural connectivity) with an increased capacity for functional flexibility. Reviewing all of the brain's neuromodulatory mechanisms, and their effects on neural dynamics, is beyond the scope of this piece (see Refs. [75–78] for reviews), as these mechanisms comprise myriad cortico-cortical, cortico-subcortical, and subcortical-subcortical interactions. Here, we focus on a specific example that we believe is well positioned to be integrated into RNNs: the balance and modulation of cortical excitation and inhibition.

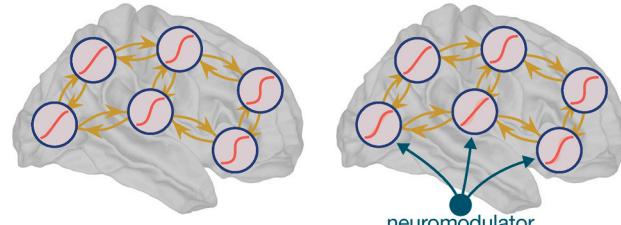
One fundamental neuromodulatory effect is that of dynamic changes to cortical excitation and inhibition. Neurons in the cortex receive a complex set of excitatory and inhibitory inputs, and the ratio between these inputs (E/I ratio) plays a critical role in coordinating an action potential. Following the S-A axis [55], the E/I ratio varies systematically across the cortex [79–81], leading to baseline differences in regions' dynamics and computation [60,82–84]. Moreover, incorporating regional variations to the E/I ratio into biophysical models has been shown to improve their fit to empirical functional data [83,85], demonstrating that the E/I ratio shapes large-scale brain dynamics. However, unlike features of brain structure that track the S-A axis [55], regions' baseline E/I ratio can be dynamically shifted via up- or down-regulating the excitatory and inhibitory neurotransmitters of post-synaptic cells [75]. This regulation is achieved via multiple neurochemical pathways which can be driven exogenously—for example, via pharmaceutical agents [86] or chemogenetics [87,88]—or endogenously, for example via the ascending noradrenergic arousal system (AAS) [75].

In dynamical systems, changes to neuronal excitation and inhibition are thought to engender population-level changes in neural gain (Fig. 3) [75]; the slope of a function that maps simulated neurons' inputs to their outputs. By tuning the neural gain between coupled oscillators, Shine

(A) Neurons with tanh activation function



(B) Neural gain varies over the cortex and is tuned by neuromodulation



(C) Neuromodulation affects dynamics

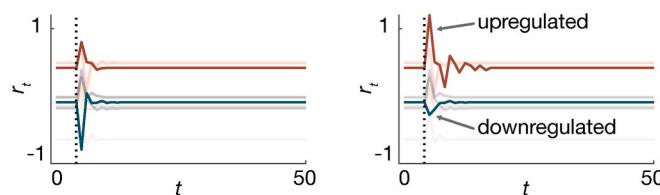


Fig. 3. Tuning neural gain in RNNs. A, Neural gain can be dynamically tuned in a trained RNN by modifying the slope of neurons' activation function. B, Neuroscientific research demonstrates that neural gain varies systematically across the cortex (left) and is tuned dynamically by neuromodulators (right). C, This neuromodulation can be used to change the dynamics of RNNs.

et al. [89] observed that increased gain lead to greater functional integration between neural populations. Critically, functional integration is thought to be an important computational property of the brain; in the human brain, functional integration fluctuates over short time scales [90] and facilitates cross-talk between the brain's many functionally-specialized communities [91]. Thus, on-the-fly changes to neurons' E/I ratio facilitates a diverse range of dynamic behaviors [11]. This diversity allows brain function to flexibly decouple from its underlying structural architecture [92–95], which in turn supports a broader range of computations than would otherwise be possible.

The above data suggests that modulation of regions' E/I ratio gives rise to state-dependent dynamics that facilitate the brain's computational expressivity. Recently, researchers have begun examining how E/I modulation might be instantiated in RNNs, with a particular focus on the aforementioned AAS. The AAS stems from the locus coeruleus, a small brainstem structure that provides diffuse noradrenergic projections spanning the cerebral cortex [75,96]. These projections modulate neuronal excitability via the neurotransmitter noradrenaline, granting the AAS the capacity to modulate the E/I ratio. Drawing on this mechanism, Wainstein et al. [97] trained an RNN to perform a perceptual switching task, wherein one visual stimuli (a plane) gradually morphed into another (a shark) and the RNN was tasked with reporting which stimuli it perceived at each time point. Once trained, Wainstein et al. [97] modified the slope of the artificial neurons' activation function (i.e., the neural gain) and examined the corresponding change in perceptual switching. The authors observed that higher gain caused perceptual switches to occur earlier than expected, while lower gain caused the opposite. Additionally, Wainstein et al. [97] modeled the energy landscape of the RNN state-space and observed that increasing neurons' gain flattened the landscape, allowing for easier state transitions (perceptual switches). Finally, the authors supported these modeling results with task-based fMRI data as well as pupillometry data, which is thought to be an indirect measure of noradrenaline-mediated arousal [98]. Together, the authors' results demonstrate that a system's computational function can be dynamically modulated in behaviorally meaningful ways, and that this reconfiguration may be underpinned by an internal capacity to regulate neural excitability. Critically, this dynamic reconfiguration unfolds on top of a static network architecture, wherein only neurons' activation functions are tweaked while their trained weights are preserved.

The results of Wainstein et al. [97] demonstrate that the affects of neuromodulation can be introduced to RNNs, modifying their functional outcomes in behaviorally meaningful ways. However, as touched on above, the brain comprises multiple neuromodulatory systems that are capable of influencing regions' excitation and inhibition, each of which subserve different functional goals [75] and each exhibit unique spatial patterning of their associated neurotransmitters and genes [80]. Future work examining how each of these neurotransmitter maps affect RNN behavior, across a diverse range of tasks, will be important to characterize how different neuromodulatory mechanisms influence expressivity. Indeed, other fields of dynamical systems (e.g. linear systems) have already begun pursuing these goals [99–101].

4. Computing with the latent spaces of RNNs via constrained connectivity

While expressivity tells us what information about the input is contained in a specific state, it does not tell us about the computational meaning behind that state. Specifically, although Eq. (7) provides us with a map of how any input series u_t is expressed as a specific neural state x_{t+1} , the meaning of that state depends on the context of the problem being solved. For example, while the dynamics of the transistors in a microprocessor can be known and simulated, the computational meaning of the transistor state depends on its internal, or *latent* representation [102].

As an illustration of latent representation, consider one of the

fundamental memory elements of computers, the *set-reset latch* (SR-latch) [103], which simply remembers which of two inputs were pulsed most recently. A single nonlinear neuron with two inputs can be designed to mimic this behavior (Fig. 4A), where its state remains high if input u_1 was last pulsed, and remains low if input u_2 was last pulsed. Here, the state of the neuron is directly the output of a latch. Alternatively, this latch functionality can be defined in a *distributed* manner into a system of multiple neurons, where the high state is represented as some pattern of activity r^* , the low state is represented as another pattern of activity r^\dagger , and the input pulses transition the RNN state between these two (Fig. 4B). Here, no single neuron is responsible for the latch dynamics. Rather, these latent-space latch dynamics depend on the connectivity between neurons, as well as how that connectivity was formed by training. In this section, we discuss how RNNs represent and manipulate information in their latent space, and the consequence of biological constraints on these latent representations. Then, we draw on recent advances from the field of neurodevelopment to put forth new directions for studying biologically-constrained RNNs.

4.1. Sparsity and attractor stability

Neural networks harness the power of internal, or *latent*, representations for computational tasks such as path integration [104], tracking [105], and spatial working memory [106]. In RNNs and dynamical systems, these latent representations are often referred to as *attractors*: sets of points, $\mathcal{S} = \{s_i\}$, to which the dynamics evolve after a relatively long period of time. An early example of latent representations is associative memory in Hopfield networks [107], wherein a specific set of neural activity patterns, $\mathcal{S} = \{s_i\}$, are stored as memories that could represent information such as an image. Specifically, \mathcal{S} are stored as *fixed-point attractors* [108] such that after stimulating neurons close to a specific pattern, $x_0 = s_i + \epsilon$, the neural states will evolve toward a stored memory, $x_{t \rightarrow \infty} \approx s_i$. In general, a fixed-point is a state x^* such that

$$x^* = f(x^*). \quad (8)$$

The computational power of this Hopfield network is in using inputs to retrieve pre-defined information (i.e., *memories*) stored in attractors, and substantial work has gone into improving their computational capability [109] and biophysical realism [110]. Hence, fixed-point attractors are latent dynamical properties that can be harnessed for computation.

In addition to discrete memory states, RNNs can make use of the *geometry* of their attractors to form representations and make decisions. For example, *continuous-attractor neural networks* (CANNs) extend the concept of an attracting point to higher-dimensional manifolds, thereby forming attracting curves and surfaces such that the geometric position along these manifolds holds latent computational meaning. For example, the geometric trajectory of the neural network state along these manifolds can reflect a path traversed in real physical space by an agent [104], the continuous tracking of a moving stimulus [105], and the recall of spatial location in the prefrontal cortex [106]. While the connectivity and dynamics of CANNs are precisely engineered to preserve translation-invariance along their structure, this continuum of attractors also emerges in trained RNN models [111], and even in models of the prefrontal cortex trained to integrate information given different contexts [3]. Hence, rather than the activity of one or a collection of specific neurons [112], it is the geometry of the attractor manifold that forms the internal representation of information in the RNN, and the RNN integrates external information by moving its representation along that manifold [113].

Formation of attractors occurs as a consequence of a loss of energy in the system, which in turn results in the stabilization of dynamics [114]. This stabilization can be characterized using methods such as Lyapunov functions—energy-like quantities that monotonically decrease or dissipate throughout the dynamics [115,116]—and Lyapunov exponents—the rate of convergence towards an attracting manifold [117]—

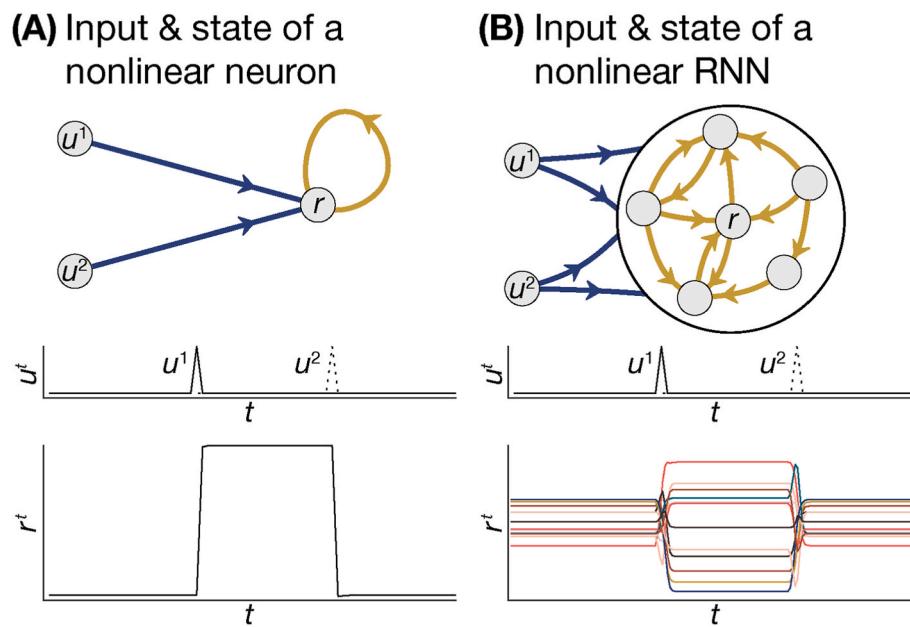


Fig. 4. Distributed latent-space representations. **A**, A nonlinear single-neuron model with two inputs (top), which has been designed to act as a memory circuit such that an impulse from input u_1 or u_2 will cause the state of the neuron to stay fixed at a high or low value, respectively (bottom). **B**, A nonlinear multi-neuron RNN which has been designed to act as a memory circuit (top), except that the RNN represents the states “high” and “low” as two stable fixed-points across all of its states, and these fixed-points are determined in a distributed manner via all of the connections between the neurons.

to study existing systems. However, what is less clear is how connectivity can develop to improve the stabilization of these attractors. Intuitively, this energy dissipation usually takes the form of a loss in the neural activity (e.g. the “leaky” component of the leaky integrator in Eq. (2)), whereas the parameters that can be learned in the RNN are the *connectivity*. Thus, the question becomes: how do we modify the RNN connectivity to achieve greater attractor stability?

A biologically-inspired optimization process that has proven useful for stabilizing RNN dynamics is *sparseness*. In order to minimize energy expenditure [114], the brain substantially prunes its connectivity [118] retaining only a sparse set of weights that are finely tuned to achieve its functional goals. In RNNs, inducing sparseness via weight pruning has been shown to provide several computational benefits. For example, Averbeck [119] trained RNNs with and without weight pruning to complete a working memory task. Compared to their unpruned counterparts, moderate amounts of pruning yielded RNNs that (i) exhibited better task performance; (ii) required fewer training epochs; (iii) had stronger connectivity weights; and (iv) were more resistant to task distractors [119]. Notably, regarding distractor resistance, pruned RNNs showed a smaller departure from their dynamic trajectories when they were perturbed by a distracting probe within the task. This result demonstrates that the sparse connectivity in the pruned RNNs strengthened their attractor basins (Fig. 5), making them more stable and resistant to undesired inputs.

4.2. Deduction: learning problem structure through iterative algorithms

While the geometry of dynamical attractors can be used for computational purposes, they also arise as the solutions to complex problems. For example, iterative methods are commonly used in optimization problems such as iterative refinement [120], root-finding methods [121], and feasibility problems [122]. Critically, in these and many other examples, the solution is not learned in the typical deep learning sense (i.e., training), but rather emerges as an attractor to satisfy the conditions of an iterative *algorithm*. In the same manner, RNNs need not only learn the attractor structure of specific input-output relations (i.e., through training), but have the potential to encode a specific algorithm in the iteration of the neural states, such that solutions

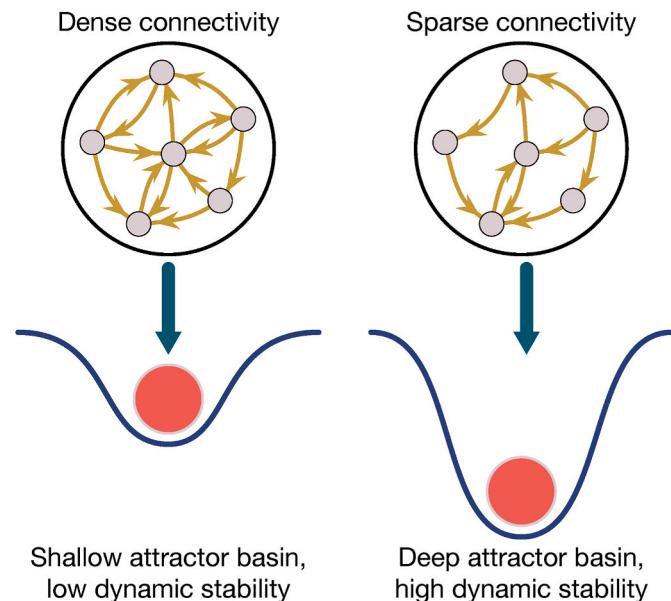


Fig. 5. Sparse connectivity leads to stable attractors. Pruning connectivity weights in RNNs leads to sparse weight matrices that deepen the attractor basins of RNNs dynamics, making dynamics more stable.

to problems (i.e., tasks) emerge as attractors.

Biological neural networks demonstrate the ability to run iterations within their latent representations [123]. A prominent example is hippocampal replay, whereby hippocampal place cells will reactivate along the same sequence as in a prior navigation experience [124], even when the subject is not actively performing a navigation task. Another prominent example is dynamical inference, whereby neural activity in the dlPFC can reliably predict the future nonlinear trajectory of a ball, and RNN models which best replicate prediction behavior are trained on the sequence of the ball’s trajectory [125,126]. Hence, RNNs are not only capable of learning attractor geometries, but also of learning and

simulating the sequence of the problem structure, which may enable more generalizable solutions.

RNNs can be engineered to run iterative deductions through several means. One approach involves assigning to each neuron the state variable of an algorithm, and defining complex interaction dynamics such that the RNN state will settle on the solution as a stable attractor. For example, an RNN can be designed to solve k -satisfiability problems, which seek an assignment of n Boolean variables that satisfy a set of c constraints, where each constraint places a condition on subsets of k Boolean variables (Fig. 6A) [127]. These RNNs evolve until the neural states find a solution [127]. Surprisingly, a wide variety of different dynamics and architectures can lead to different algorithms for solving the same satisfiability problem [128,129], and other difficult optimization problems such as integer linear programming feasibility [130] or the n -queens problem [131]. Hence, algorithm variables can be directly represented by individual neurons, and the algorithm rules can be directly encoded in the connectivity and update rules f of RNNs.

In addition to a direct, one-to-one encoding of problem structure as an RNN, we can also design algorithms into the latent spaces of RNNs [132]. Rather than ascribing each neuron a specific variable in an algorithm, we can embed an algorithm into the distributed connectivity of an RNN (Fig. 6B). One technique for this embedding is the *neural engineering framework* (NEF) which enables the design of iterations of the latent-space variables through the engineering of low-rank connectivity matrices [133,134]. Extensions enable the programming of iterations in pre-existing, higher-rank connectivity, and the ability to reverse-engineer representations from conventionally trained RNNs [36]. Other engineered architectures such as the differentiable neural computer [135] and the neural turing machine [136] emulate the structure of conventional computers using differentiable neural elements. Hence, RNNs have the capability to explicitly run complex algorithms in their latent spaces, which is crucial for generalizable computation; rather than learning individual solutions, we posit that training RNNs on solution sequences will enable them to learn and generalize problem-solving strategies [137], even into nonlinear, out-of-sample regimes [138].

While the above approaches may provide more generalizable RNN solutions than task-specific training, the added computational capabilities of engineered RNNs is accompanied by a further deviation of the corresponding connectivity from biology. Whether through the enforcing of low-rank connectivity [133,139] or the segregation of memory and processing units in a neural von Neumann architecture [135], engineered neural connectivities lack many of the costs and constraints experienced by biological networks. A critical question then is how the brain formulates connectivities that permit sophisticated latent space computations while adhering to biological constraints. Prior work has demonstrated the capability of largely disordered RNNs to produce sequences that rely on recurrent connectivity [140], and the importance of the sequence of learning over many learning iterations—a curriculum—for RNN performance [141]. Hence, rather than forming low-rank structures *ab initio*, the brain defines its structure and dynamics through learning and plasticity on long time scales. We explore insights into the governing principles and computational advantages of this progression in neurodevelopment.

4.3. Latent-space computational capability throughout neurodevelopment

Just as the topology of an RNN is sculpted over training epochs (Fig. 7A), the topology of the human connectome is sculpted throughout development (Fig. 7B). However, unlike the RNN, which may be trained in an unconstrained manner, neurodevelopment follows a carefully orchestrated and stereotyped program that unfolds dynamically across space and time. Specifically, cortical neurodevelopment is thought to spatially track the aforementioned S-A axis in a temporally staged manner [55,142], and this staging is thought to underpin the emergence of cortical regions' functional specialization and inter-connectivity. Crucially, the asynchronous nature of this developmental program is thought to underpin the sequential emergence of increasingly complex cognitive functions [143,144], suggesting that neurodevelopment stages the brain's acquisition of lower- and higher-order computational processes. Mechanistically, this program may be underpinned by windows of heightened neural plasticity that cascade up the S-A axis [142,145]

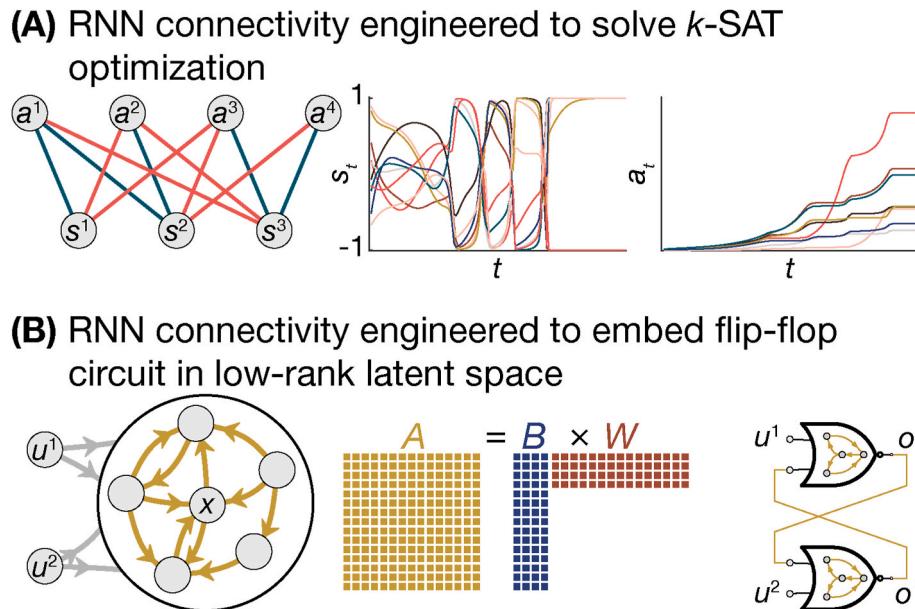


Fig. 6. Programming Algorithms into RNNs **A**, Schematic of an RNN whose neurons s_i and a_j represent states in a k -SAT problem, and whose connectivity is defined such that the stable attracting manifold is guaranteed to be a solution (left). The dynamics of the RNN evolve in a complex manner as it searches and finds the assignment of states that satisfies the k -SAT problem (center, right). **B**, Schematic of the process for designing algorithms into RNNs, where the recurrent connectivity of the RNN (gold) can be constructed from a low-rank product of matrices (B, W). Through this construction, the evolution of the RNN states concurrently evolves the state variables of the algorithm within a low-dimensional projection. This designed connectivity was used to program and simulate the SR-latch in Fig. 4B). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

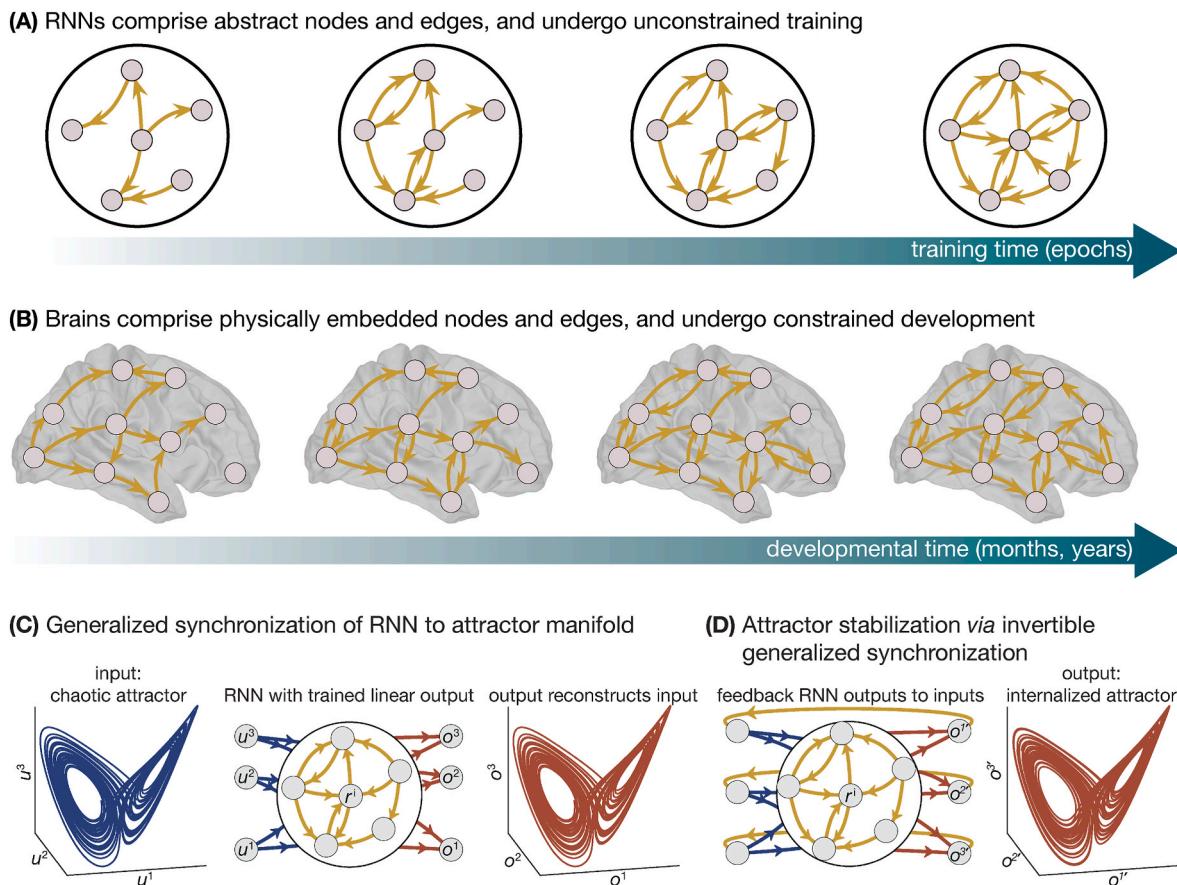


Fig. 7. Computational capabilities of RNNs and biological brains throughout development. RNNs and brains are subject to different environments. **A**, RNNs comprise abstract nodes and edges, and their training is unconstrained. This allows RNNs to evolve toward any network architecture that optimally performs a given task. **B**, By contrast, brains comprise physically embedded nodes and edges, and the formations of their network architecture is tightly constrained by space and time. **C**, The extent to which pre-existing connectivities can follow latent representations can be captured through *generalized synchronization*, which measures how well an RNN that is driven by an attracting manifold (in this case, the chaotic Lorenz attractor) can reconstruct the driving signal from its states. **D**, The ability of pre-existing connectivities to *internalize* and generate latent representations is captured through *invertible generalized synchronization*, whereby an RNN that has learned output weights W that copy the attractor manifold can then *drive itself* by feeding those outputs back as inputs to autonomously generate the attractor *without any input*.

priming specific neural circuits at specific points in time for experience-dependent neural change.

Regions in the cortex are defined in part according their laminar structure [146,147], with different regions exhibiting variations in the number and size of their distinct layers, as well as different distributions of cells throughout those layers. Critically, cortical variations in cytoarchitecture conform to the S-A axis [148], and animal research demonstrates that this spatial patterning predicts regions' extrinsic connectivity [149], including their strength [150], distance [150], and layer-wise projections [150,151]. In humans, structural connectivity between regions at the bottom of the S-A axis refines relatively early in development, while connectivity at the top of the S-A axis does so later in development [74,152–155]. Furthermore, recent work has shown that cytoarchitecture plays an important role in shaping how dynamics spread across the connectome throughout development [72]. Thus, the spatial patterning embedded in the S-A axis plays a key role in shaping connectome topology throughout development. But what about RNNs? Recent work by Achterberg et al. [156] regularized RNNs by using the Euclidean distance between regions to constrain training. The authors found that modularity [157] and small-worldness [158]—two complex topological features that are hallmarks of the human connectome [28]—emerged to a greater extent in these spatially-embedded RNNs compared to standard RNNs (see also recent work by Tanner et al. [159] for evidence of modularity in RNNs trained without spatial constraints). Additionally, this effect coincided with achieving higher out-of-sample task performance earlier in training compared to standard RNNs

(though performance eventually converged; see their Fig. 2A).

The results of Achterberg et al. [156] demonstrate that incorporating space-based inductive biases into RNN training causes them to converge on topological features observed in the human connectome. However, it remains unclear whether neurodevelopmentally-informed spatial constraints, like those embedded in the S-A axis [55], show similar effects. We posit that constraining RNNs using the S-A axis may outperform Euclidean distance-based spatial embedding, as the former is rooted in evolutionary programs of connectivity formation and functional specialization [57]. Additionally, the spatial constraints deployed by Achterberg et al. [156] were static throughout training. As mentioned above, the S-A axis scaffolds connectome development in a temporally varying way [142], and incorporating this dynamic information will be critical to achieving realistic brain-like topology in RNNs. One approach would be to code spatially varying periods of heightened learning into RNNs, simulating traveling waves of heightened neural plasticity [142, 145]. Such an inductive bias could be achieved by including temporal cascades of weight training that flow bottom-up across the S-A axis.

In addition to injecting spatially constrained inductive biases into RNNs, we can also directly assess the ability of the developing connectome to support latent-space computation. This analysis can be achieved through studying the *synchronization* between a given RNN and a particular latent attractor. Historically, synchronization has been shown to be crucial for computation in both biological cortical networks [160] and artificial RNNs [161], and is deeply related to consensus dynamics [162]. Intuitively, synchronization between two systems implies

that both systems are evolving identically. This concept can be extended to *generalized synchronization*, which stipulates conditions under which a response system, r_t , has synchronized in a general sense to a driving system, d_t [163], such that rather than evolving identically such that $r_t = d_t$ [164], the joint system has collapsed onto a function φ of only the driver system such that $r_t = \varphi(d_t)$. Under these conditions, the response system has *followed* the attractor structure of the driving system. If we choose the response system to be an RNN with an empirically-derived connectivity taken at a specific point in neurodevelopment, and the driver system to be a specific latent attractor dynamical system, then we can assess whether the RNN can follow the attractor structure of the driving system (Fig. 7C).

Of course, just because the RNN can synchronize to a particular latent-space attractor does not guarantee that it can maintain that attractor once the driving system is gone. In order for the RNN to *internalize* the attractor dynamics, theories of *invertible generalized synchronization* (IGS) stipulate conditions for which the attractor of the driver system can be invertibly reproduced and stabilized by the response system, and thus can be *learned* autonomously [35]. Hence, rather than modifying RNN connectivity to learn a latent space conditioned on spatial and resource constraints, IGS tests whether a given RNN connectivity that already obeys those spatial and resource constraints can stably generate a latent attractor (Fig. 7D). The IGS theory also indicates that the ability to internalize latent attractors from driving systems depends not only on the RNN's structure, but also the latent attractor. Thus, given RNNs of different structures throughout development, we may examine the IGS on their most-likely encountered driving signals corresponding to their position along the S-A axis.

5. Conclusions

Neural systems compute using dynamics, and the dynamics of biological brains evolve atop the computational substrate of a spatially and resource constrained network. Here, we sought to jointly discuss advances in neuroscience and dynamical systems with a view to improving the computational power and interpretability of RNNs. Within the context of two computational capabilities—expressivity and latent-space computing—we highlighted several avenues for future research that we believe will advance our understanding of dynamical computation. Through these avenues, we envision biologically interpretable, computationally improved RNN models of how the brain computes.

Central to this research program is the application of biophysical constraints on the computational dynamics of RNNs. While not exhaustive, the constraints discussed herein represent the diverse influences of neurobiology on network structure, and they have been shown to influence the emergence of complex behavior in humans. Furthermore, the influence of these constraints on RNN connectivity can be studied in combination. For example, the spatial-patterning of regions' baseline E/I ratio emerges throughout development [142], indicating its connection to the S-A axis. Another example is to explore how to extend low-rank RNN design approaches to higher-rank connectivities that more closely match the spatial gradients of the S-A axis. Thus, these biophysical constraints provide fertile ground for future experimental, computational, and theoretical work into biologically-informed RNNs.

The methodological strategies for incorporating these constraints into computational RNN modeling are vast. Here, we have discussed several approaches: using known cortical structure and function as target connectivities of additional training constraints [55,156], using resource constraints as an alternative learning mechanism [32], dynamically altering connectivity with neuromodulation for increased expressivity [89], stabilizing latent attractors through pruning [119], engineering and modifying low-rank latent representations and algorithms [36,133], probing the teachability of RNNs *via* synchronization [35], among many others. Hence, there is no one prescription that serves as a panacea for the rich problems that lie at the intersection of computation, dynamics, and neurobiology. Instead, we must continue

developing diverse and creative approaches for maximizing the computational capabilities of neurodynamical models through the use of biological and developmental constraints.

CRediT authorship contribution statement

Jason Z. Kim: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Bart Larsen:** Writing – review & editing, Writing – original draft. **Linden Parkes:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We gratefully acknowledge comments and conversations with Dr. Zhixin Lu, Dr. Harang Ju, Dr. Dale Zhou, Dr. Melody Lim, and Dr. Stephen Hanson. JZK was supported by the Bethe/KIC/Wilkins post-doctoral fellowship. BL was supported by the National Institute Of Mental Health of the National Institutes of Health under Award Number R00MH127293. LP was supported by the National Institute Of Mental Health of the National Institutes of Health under Award Number R00MH127296.

References

- [1] A.J. Ijspeert, Central pattern generators for locomotion control in animals and robots: a review, *Neural Network*. 21 (2008) 642–653.
- [2] W.W. Seeley, et al., Dissociable intrinsic connectivity networks for salience processing and executive control, *J. Neurosci.* 27 (2007) 2349–2356.
- [3] V. Mante, D. Sussillo, K.V. Shew, W.T. Newsome, Context-dependent computation by recurrent dynamics in prefrontal cortex, *nature* 503 (2013) 78–84.
- [4] P.J. Werbos, Backpropagation through time: what it does and how to do it, *Proc. IEEE* 78 (1990) 1550–1560.
- [5] R.F. Betzel, et al., Generative models of the human connectome, *Neuroimage* 124 (2016) 1054–1064.
- [6] S. Oldham, A. Fornito, The development of brain network hubs, *Dev. Cognit. Neurosci.* 36 (2019) 100607.
- [7] S. Oldham, et al., Modeling spatial, developmental, physiological, and topological constraints on human brain connectivity, *Sci. Adv.* 8 (2022) eabm6127.
- [8] D. Akarca, et al., A generative network model of neurodevelopmental diversity in structural brain organization, *Nat. Commun.* 12 (2021) 4216.
- [9] A.L. Hodgkin, A.F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve, *J. Physiol.* 117 (1952) 500.
- [10] K.-E. Petousakis, A.A. Apostolopoulou, P. Poirazi, The impact of hodgkin-huxley models on dendritic research, *J. Physiol.* (2022).
- [11] H.R. Wilson, J.D. Cowan, Excitatory and inhibitory interactions in localized populations of model neurons, *Biophys. J.* 12 (1972) 1–24.
- [12] D.J. Pinto, J.C. Brumberg, D.J. Simons, G.B. Ermentrout, R. Traub, A quantitative population model of whisker barrels: re-examining the wilson-cowan equations, *J. Comput. Neurosci.* 3 (1996) 247–264.
- [13] S. Sadeghi, D. Mier, M.F. Gerchen, S.N. Schmidt, J. Hass, Dynamic causal modeling for fmri with wilson-cowan-based neuronal equations, *Front. Neurosci.* 14 (2020) 593867.
- [14] L. Parkes, et al., Using network control theory to study the dynamics of the structural connectome, preprint, *Neuroscience* (2023).
- [15] C.W. Lynn, D.S. Bassett, The physics of brain network structure, function and control, *Nature Rev. Phys.* 1 (2019) 318–332.
- [16] C. Seguin, O. Sporns, A. Zalesky, Brain network communication: concepts, models and applications, *Nat. Rev. Neurosci.* (2023).
- [17] P. Srivastava, et al., Models of communication and control for brain networks: distinctions, convergence, and future outlook, *Network Neurosci.* 4 (2020) 1122–1159.
- [18] M. Breakspear, Dynamic models of large-scale brain activity, *Nat. Neurosci.* 20 (2017) 340–352.

- [19] J.A. Roberts, et al., Metastable brain waves, *Nat. Commun.* 10 (2019) 1056.
- [20] R. FitzHugh, Impulses and physiological states in theoretical models of nerve membrane, *Biophys. J.* 1 (1961) 445–466.
- [21] W.S. McCulloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, *Bull. Math. Biophys.* 5 (1943) 115–133.
- [22] S. Chung, H. Siegelmann, Turing completeness of bounded-precision recurrent neural networks, *Adv. Neural Inf. Process. Syst.* 34 (2021) 28431–28441.
- [23] Y. Liu, C. Gong, L. Yang, Y. Chen, Dstprnn: a dual-stage two-phase attention-based recurrent neural network for long-term and multivariate time series prediction, *Expert Syst. Appl.* 143 (2020) 113082.
- [24] Z. Lu, J.Z. Kim, D.S. Bassett, Supervised chaotic source separation by a tank of water, *Chaos: Interdisciplin. J. Nonlinear Sci.* 30 (2020).
- [25] P.Y. Wang, Y. Sun, R. Axel, L. Abbott, G.R. Yang, Evolving the olfactory system with machine learning, *Neuron* 109 (2021) 3879–3892.e5.
- [26] G.R. Yang, M.R. Joglekar, H.F. Song, W.T. Newsome, X.-J. Wang, Task representations in neural networks trained to perform many cognitive tasks, *Nat. Neurosci.* 22 (2019) 297–306.
- [27] J.A. Roberts, et al., The contribution of geometry to the human connectome, *Neuroimage* 124 (2016) 379–393.
- [28] E. Bullmore, O. Sporns, The economy of brain network organization, *Nat. Rev. Neurosci.* 13 (2012) 336–349.
- [29] S. Ramón y Cajal, *Histology of the nervous system of man and vertebrates*. No. No. 6 in History of Neuroscience, Oxford University Press, New York, 1995.
- [30] E. Dupont, A. Doucet, Y.W. Teh, Augmented neural odes, *Adv. Neural Inf. Process. Syst.* 32 (2019).
- [31] M. Lukoševičius, H. Jaeger, Reservoir computing approaches to recurrent neural network training, *Comput. Sci. Rev.* 3 (2009) 127–149.
- [32] A. Ali, N. Ahmad, E. de Groot, M.A.J. van Gerven, T.C. Kietzmann, Predictive coding is a consequence of energy efficiency in recurrent neural networks, *Patterns* 3 (2022).
- [33] H. Jaeger, Tutorial on Training Recurrent Neural Networks, Covering Bppt, Rtrl, Elk and the " Echo State Network" Approach, 2002.
- [34] L.E. Suárez, et al., conn2res: a toolbox for connectome-based reservoir computing, *bioRxiv* 2023–05 (2023).
- [35] Z. Lu, D.S. Bassett, Invertible generalized synchronization: a putative mechanism for implicit learning in neural systems, *Chaos: Interdisciplin. J. Nonlinear Sci.* 30 (2020).
- [36] J.Z. Kim, D.S. Bassett, A neural machine code and programming framework for the reservoir computer, *Nat. Mach. Intell.* 1–9 (2023).
- [37] M. Felleisen, On the expressive power of programming languages, *Sci. Comput. Program.* 17 (1991) 35–75.
- [38] M. Sipser, Introduction to the theory of computation, *ACM Sigact News* 27 (1996) 27–29.
- [39] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, *Neural Network*. 2 (1989) 359–366.
- [40] G. Cybenko, Approximation by superpositions of a sigmoidal function, *Math. Control, Signals, Syst.* 2 (1989) 303–314.
- [41] A.M. Schäfer, H.G. Zimmermann, Recurrent neural networks are universal approximators, in: *Artificial Neural Networks-ICANN 2006: 16th International Conference, Athens, Greece, September 10-14, 2006. Proceedings, Part I*, 16, Springer, 2006, pp. 632–640.
- [42] B. Poole, S. Lahiri, M. Raghu, J. Sohl-Dickstein, S. Ganguli, Exponential expressivity in deep neural networks through transient chaos, *Adv. Neural Inf. Process. Syst.* 29 (2016).
- [43] M. Raghu, B. Poole, J. Kleinberg, S. Ganguli, J. Sohl-Dickstein, On the expressive power of deep neural networks. In *International Conference on Machine Learning*, PMLR, 2017, pp. 2847–2854.
- [44] N. Bertschinger, T. Natschläger, Real-time computation at the edge of chaos in recurrent neural networks, *Neural Comput.* 16 (2004) 1413–1436.
- [45] H. Sompolinsky, A. Crisanti, H.-J. Sommers, Chaos in random neural networks, *Phys. Rev. Lett.* 61 (1988) 259.
- [46] K. Rajan, L.F. Abbott, Eigenvalue spectra of random matrices for neural networks, *Phys. Rev. Lett.* 97 (2006) 188104.
- [47] C.G. Langton, Computation at the edge of chaos: phase transitions and emergent computation, *Phys. Nonlinear Phenom.* 42 (1990) 12–37.
- [48] D. Sussillo, L.F. Abbott, Generating coherent patterns of activity from chaotic neural networks, *Neuron* 63 (2009) 544–557.
- [49] M.A. Munoz, Colloquium: criticality and dynamical scaling in living systems, *Rev. Mod. Phys.* 90 (2018) 031001.
- [50] J. Hochstetter, et al., Avalanches and edge-of-chaos learning in neuromorphic nanowire networks, *Nat. Commun.* 12 (2021) 4008.
- [51] H. Ju, J.Z. Kim, J.M. Beggs, D.S. Bassett, Network structure of cascading neural systems predicts stimulus propagation and recovery, *J. Neural. Eng.* 17 (2020) 056045.
- [52] T.L. Carroll, Optimizing memory in reservoir computers, *Chaos: Interdisciplin. J. Nonlinear Sci.* 32 (2022).
- [53] A. Wolff, et al., Intrinsic neural timescales: temporal integration and segregation, *Trends Cognit. Sci.* (2022) S1364661321002928.
- [54] M. Mesulam, Representation, inference, and transcendent encoding in neurocognitive networks of the human brain, *Ann. Neurol.* 64 (2008) 367–378.
- [55] V.J. Sydnor, et al., Neurodevelopment of the association cortices: patterns, mechanisms, and implications for psychopathology, *Neuron* 109 (2021) 2820–2846.
- [56] D.S. Margulies, et al., Situating the default-mode network along a principal gradient of macroscale cortical organization, *Proc. Natl. Acad. Sci. USA* 113 (2016) 12574–12579.
- [57] T. Xu, et al., Cross-species functional alignment reveals evolutionary hierarchy within the connectome, *Neuroimage* 223 (2020) 117346.
- [58] U. Hasson, E. Yang, I. Vallines, D.J. Heeger, N. Rubin, A hierarchy of temporal receptive windows in human cortex, *J. Neurosci.* 28 (2008) 2539–2550.
- [59] R. Gao, R.L. van den Brink, T. Pfeffer, B. Voytek, Neuronal timescales are functionally dynamic and shaped by cortical microarchitecture, *Elife* 9 (2020) e61277.
- [60] V.J. Sydnor, et al., Intrinsic activity development unfolds along a sensorimotor-association cortical axis in youth, *Nat. Neurosci.* 26 (2023) 638–649.
- [61] J.P. Hespanha, *Linear Systems Theory*, Princeton university press, 2018.
- [62] A.H. Marblestone, G. Wayne, K.P. Kording, Toward an integration of deep learning and neuroscience, *Front. Comput. Neurosci.* 10 (2016).
- [63] A. Ali, N. Ahmad, E. De Groot, M.A. Johannes Van Gerven, T.C. Kietzmann, Predictive coding is a consequence of energy efficiency in recurrent neural networks, *Patterns* 3 (2022) 100639.
- [64] B. Millidge, A. Seth, C.L. Buckley, Predictive Coding: a Theoretical and Experimental Review, 2021. Publisher: arXiv Version Number: 4.
- [65] A.M. Bastos, et al., Canonical Microcircuits for predictive coding, *Neuron* 76 (2012) 695–711.
- [66] W. Singer, Recurrent dynamics in the cerebral cortex: integration of sensory evidence with stored knowledge, *Proc. Natl. Acad. Sci. USA* 118 (2021) e2101043118.
- [67] K. Friston, Computational psychiatry: from synapses to sentience, *Mol. Psychiatr.* (2022).
- [68] G. Tanaka, et al., Recent advances in physical reservoir computing: a review, *Neural Network*. 115 (2019) 100–123.
- [69] G. Van der Sande, D. Brunner, M.C. Soriano, Advances in photonic reservoir computing, *Nanophotonics* 6 (2017) 561–576.
- [70] M.C. Soriano, et al., Delay-based reservoir computing: noise effects in a combined analog and digital implementation, *IEEE Transact. Neural Networks Learn. Syst.* 26 (2014) 388–393.
- [71] L.E. Suárez, B.A. Richards, G. Lajoie, B. Misic, Learning function from structure in neuromorphic networks, *Nat. Mach. Intell.* (2021).
- [72] L. Parkes, et al., Asymmetric signaling across the hierarchy of cytoarchitecture within the human connectome, *Sci. Adv.* 8 (2022) eadd2185.
- [73] A. Pines, et al., Development of top-down cortical propagations in youth, *Neuron* (2023) S0896627323000387.
- [74] G.L. Baum, et al., Modular segregation of structural brain networks supports the development of executive function in youth, *Curr. Biol.* 27 (2017) 1561–1572.e8.
- [75] J.M. Shine, et al., Computational models link cellular mechanisms of neuromodulation to large-scale neural dynamics, *Nat. Neurosci.* 24 (2021) 765–776.
- [76] E. Marder, V. Thirumalai, Cellular, synaptic and network effects of neuromodulation, *Neural Network*. 15 (2002) 479–493.
- [77] D. Bucher, J.-M. Goaillard, Beyond faithful conduction: short-term dynamics, neuromodulation, and long-term regulation of spike propagation in the axon, *Prog. Neurobiol.* 94 (2011) 307–346.
- [78] D.A. McCormick, D.B. Nestvogel, B.J. He, Neuromodulation of brain state and behavior, *Annu. Rev. Neurosci.* 43 (2020) 391–415.
- [79] Y. Kim, et al., Brain-wide maps reveal stereotyped Cell-type-based cortical architecture and subcortical Sexual Dimorphism, *Cell* 171 (2017) 456–469.e22.
- [80] J.B. Burt, et al., Hierarchy of transcriptomic specialization across human cortex captured by structural neuroimaging topography, *Nat. Neurosci.* 21 (2018) 1251–1259.
- [81] K.M. Anderson, et al., Transcriptional and imaging-genetic association of cortical interneurons, brain function, and schizophrenia risk, *Nat. Commun.* 11 (2020) 2889.
- [82] G. Deco, et al., How local excitation-inhibition ratio impacts the whole brain dynamics, *J. Neurosci.* 34 (2014) 7886–7898.
- [83] G. Deco, et al., Dynamical consequences of regional heterogeneity in the brain's transcriptional landscape, *Sci. Adv.* 7 (2021) eabf4752.
- [84] R. Gao, E.J. Peterson, B. Voytek, Inferring synaptic excitation/inhibition balance from field potentials, *Neuroimage* 158 (2017) 70–78.
- [85] S. Zhang, et al., In-vivo whole-cortex estimation of excitation-inhibition ratio indexes cortical maturation and cognitive ability in youth. preprint, *bioRxiv* (2023).
- [86] B. Larsen, et al., A developmental reduction of the excitation:inhibition ratio in association cortex during adolescence, *Sci. Adv.* 8 (2022) eabj8750.
- [87] F. Rocchi, et al., Increased fMRI connectivity upon chemogenetic inhibition of the mouse prefrontal cortex, *Nat. Commun.* 13 (2022) 1056.
- [88] M. Markicevic, et al., Cortical excitation:inhibition Imbalance causes Abnormal brain network dynamics as observed in neurodevelopmental Disorders, *Cerebr. Cortex* 30 (2020) 4922–4937.
- [89] J.M. Shine, M.J. Auburn, M. Breakspear, R.A. Poldrack, The modulation of neural gain facilitates a transition between functional segregation and integration in the brain, *Elife* 7 (2018) e31130.
- [90] J.M. Shine, et al., Human cognition involves the dynamic integration of neural activity and neuromodulatory systems, *Nat. Neurosci.* 22 (2019) 289–296.
- [91] M.A. Bertolero, B.T.T. Yeo, M. D'Esposito, The modular and integrative functional architecture of the human brain, *Proc. Natl. Acad. Sci. USA* 112 (2015).
- [92] B. Vázquez-Rodríguez, et al., Gradients of Structure-Function Tethering across Neocortex, 116, *Proceedings of the National Academy of Sciences*, 2019, pp. 21219–21227.
- [93] M.G. Preti, D. Van De Ville, Decoupling of brain function from structure reveals regional behavioral specialization in humans, *Nat. Commun.* 10 (2019) 4747.

- [94] B. Mišić, et al., Network-level structure-function relationships in human Neocortex, *Cerebr. Cortex* 26 (2016) 3285–3296.
- [95] F. Zamani Esfahlani, J. Faskowitz, J. Slack, B. Mišić, R.F. Betzel, Local structure-function relationships in human brain networks across the lifespan, *Nat. Commun.* 13 (2022) 2053.
- [96] E. Samuels, E. Szabadi, Functional Neuroanatomy of the noradrenergic locus caeruleus: its roles in the regulation of arousal and autonomic function Part II: physiological and Pharmacological manipulations and Pathological Alterations of locus caeruleus activity in humans, *Curr. Neuropharmacol.* 6 (2008) 254–285.
- [97] G. Wainstein, et al., Gain neuromodulation mediates perceptual switches: evidence from pupillometry, fMRI, and RNN Modelling. preprint, Res. Square (2023).
- [98] S. Joshi, J.I. Gold, Pupil size as a window on neural substrates of cognition, *Trends Cognit. Sci.* 24 (2020) 466–480.
- [99] S.P. Singleton, et al., Receptor-informed network control theory links LSD and psilocybin to a flattening of the brain's control energy landscape, *Nat. Commun.* 13 (2022) 5812.
- [100] A.I. Luppi, et al., Transitions between cognitive topographies: contributions of network structure, neuromodulation, and disease. preprint, bioRxiv (2023).
- [101] K. Aitken, S. Mihalas, Neural population dynamics of computing with synaptic modulations, *Elife* 12 (2023) e83035.
- [102] E. Jonas, K.P. Kording, Could a neuroscientist understand a microprocessor? *PLoS Comput. Biol.* 13 (2017) e1005268.
- [103] N. Torii, et al., Asic implementation of random number generators using sr latches and its evaluation, *EURASIP J. Inf. Secur.* 2016 (2016) 1–12.
- [104] A. Samsonovich, B.L. McNaughton, Path integration and cognitive mapping in a continuous attractor neural network model, *J. Neurosci.* 17 (1997) 5900–5920.
- [105] C.A. Fung, K.M. Wong, S. Wu, A moving bump in a continuous manifold: a comprehensive study of the tracking dynamics of continuous attractor neural networks, *Neural Comput.* 22 (2010) 752–792.
- [106] K. Wimmer, D.Q. Nykamp, C. Constantinidis, A. Compte, Bump attractor dynamics in prefrontal cortex explains behavioral precision in spatial working memory, *Nat. Neurosci.* 17 (2014) 431–439.
- [107] J.J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proc. Natl. Acad. Sci. USA* 79 (1982) 2554–2558.
- [108] S.H. Strogatz, Nonlinear Dynamics and Chaos with Student Solutions Manual: with Applications to Physics, Biology, Chemistry, and Engineering, CRC press, 2018.
- [109] H. Ramsauer, et al., Hopfield Networks Is All You Need, 2020 arXiv preprint arXiv:2008.02217.
- [110] A. Storkey, Increasing the capacity of a hopfield network without sacrificing functionality, in: Artificial Neural Networks—Icann'97: 7th International Conference Lausanne, Switzerland, October 8–10, 1997 Proceedings 7, Springer, 1997, pp. 451–456.
- [111] L.M. Smith, J.Z. Kim, Z. Lu, D.S. Bassett, Learning continuous chaotic attractors with a reservoir computer, *Chaos: Interdisciplin. J. Nonlinear Sci.* 32 (2022).
- [112] M.J. Nichols, W.T. Newsome, Middle temporal visual area microstimulation influences veridical judgments of motion direction, *J. Neurosci.* 22 (2002) 9530–9540.
- [113] M. Jazayeri, S. Ostožić, Interpreting neural computations by examining intrinsic and embedding dimensionality of neural activity, *Curr. Opin. Neurobiol.* 70 (2021) 113–120.
- [114] C. Howarth, P. Gleeson, D. Attwell, Updated energy Budgets for neural computation in the Neocortex and Cerebellum, *J. Cerebr. Blood Flow Metabol.* 32 (2012) 1222–1232.
- [115] P. Giesl, S. Hafstein, Review on computational methods for lyapunov functions, *Discr. Continuous Dynam. Syst.-B* 20 (2015) 2291–2331.
- [116] R. Bellman, Vector lyapunov functions, *J. Soc. Ind. Appl. Math. Control* 1 (1962) 32–34.
- [117] A. Wolf, J.B. Swift, H.L. Swinney, J.A. Vastano, Determining lyapunov exponents from a time series, *Phys. Nonlinear Phenom.* 16 (1985) 285–317.
- [118] Z. Petanjek, et al., Extraordinary Neoteny of Synaptic Spines in the Human Prefrontal Cortex, 108, Proceedings of the National Academy of Sciences, 2011, pp. 13281–13286.
- [119] B.B. Averbeck, Pruning Recurrent Neural Networks Replicates Adolescent Changes in Working Memory and Reinforcement Learning, 119, Proceedings of the National Academy of Sciences, 2022 e2121331119.
- [120] C.B. Moler, Iterative refinement in floating point, *J. ACM* 14 (1967) 316–321.
- [121] T.J. Ypma, Historical development of the Newton–raphson method, *SIAM Rev.* 37 (1995) 531–551.
- [122] F.J. Aragón Artacho, R. Campoy, M.K. Tam, The douglas–rachford algorithm for convex and nonconvex feasibility problems, *Math. Methods Oper. Res.* 91 (2020) 201–240.
- [123] T. Flesch, K. Juechems, T. Dumbalska, A. Saxe, C. Summerfield, Orthogonal representations for robust context-dependent task performance in brains and neural networks, *Neuron* 110 (2022) 1258–1270.
- [124] A.K. Gillespie, et al., Hippocampal replay reflects specific past experiences rather than a plan for subsequent choice, *Neuron* 109 (2021) 3149–3163.
- [125] R. Rajalingham, H. Sohn, M. Jazayeri, Dynamic tracking of objects in the macaque dorsomedial frontal cortex, bioRxiv 2022–06 (2022).
- [126] R. Rajalingham, A. Piccato, M. Jazayeri, Recurrent neural networks with explicit representation of dynamic latent variables can mimic behavioral patterns in a physical inference task, *Nat. Commun.* 13 (2022) 5865.
- [127] M. Ercsey-Ravasz, Z. Toroczkai, Optimization hardness as transient chaos in an analog approach to constraint satisfaction, *Nat. Phys.* 7 (2011) 966–970.
- [128] B. Molnar, Z. Toroczkai, M. Ercsey-Ravasz, Continuous-time neural networks without local traps for solving boolean satisfiability, in: 2012 13th International Workshop on Cellular Nanoscale Networks and Their Applications, IEEE, 2012, pp. 1–6.
- [129] H. Yamashita, H. Suzuki, Z. Toroczkai, K. Aihara, Bounded continuous-time satisfiability solver, in: International Symposium on Nonlinear Theory and its Applications (NOLTA2019), 2019.
- [130] C. Li, B.J. MacLennan, Continuous-time Systems for Solving 0-1 Integer Linear Programming Feasibility Problems, 2019 arXiv preprint arXiv:1905.04612.
- [131] Y. Ding, Y. Li, M. Xiao, Q. Wang, D. Li, A high order neural network to solve n-queens problem, in: The 2010 International Joint Conference on Neural Networks (IJCNN), IEEE, 2010, pp. 1–6.
- [132] F. Mastrogioviseppé, S. Ostožić, Linking connectivity, dynamics, and computations in low-rank recurrent neural networks, *Neuron* 99 (2018) 609–623.
- [133] C. Eliasmith, C.H. Anderson, Neural Engineering: Computation, Representation, and Dynamics in Neurobiological Systems, MIT press, 2003.
- [134] T. De Poli, P. Jaworski, C. Eliasmith, Nengo and low-power ai hardware for robust, embedded neurorobotics, *Front. Neurorob.* 14 (2020) 568359.
- [135] A. Graves, et al., Hybrid computing using a neural network with dynamic external memory, *Nature* 538 (2016) 471–476.
- [136] J. Tkacík, P. Kordík, Neural turing machine for sequential learning of human mobility patterns, in: 2016 International Joint Conference on Neural Networks (IJCNN), IEEE, 2016, pp. 2790–2797.
- [137] A. Fawzi, et al., Discovering faster matrix multiplication algorithms with reinforcement learning, *Nature* 610 (2022) 47–53.
- [138] J.Z. Kim, Z. Lu, E. Nozari, G.J. Pappas, D.S. Bassett, Teaching recurrent neural networks to infer global temporal structure from local examples, *Nat. Mach. Intell.* 3 (2021) 316–323.
- [139] A. Valente, S. Ostožić, J.W. Pillow, Probing the relationship between latent linear dynamical systems and low-rank recurrent neural network models, *Neural Comput.* 34 (2022) 1871–1892.
- [140] K. Rajan, C.D. Harvey, D.W. Tank, Recurrent network models of sequence generation and memory, *Neuron* 90 (2016) 128–142.
- [141] D. Kepple, R. Engelken, K. Rajan, Curriculum learning as a tool to uncover learning principles in the brain, in: International Conference on Learning Representations, 2022.
- [142] B. Larsen, V.J. Sydnor, A.S. Keller, B.T. Yeo, T.D. Satterthwaite, A critical period plasticity framework for the sensorimotor–association axis of cortical neurodevelopment, *Trends Neurosci.* (2023) S0166223623001674.
- [143] N.Y. Kim, S. Kastner, A biased competition theory for the developmental cognitive neuroscience of visuo-spatial attention, *Curr. Opin. Psychol.* 29 (2019) 219–228.
- [144] B. Tervo-Clemmens, et al., A canonical trajectory of executive function maturation during the transition from adolescence to Adulthood. Preprint, PsyArXiv (2022).
- [145] B. Larsen, B. Luna, Adolescence as a neurobiological critical period for the development of higher-order cognition, *Neurosci. Biobehav. Rev.* 94 (2018) 179–195.
- [146] M.Á. García-Cabezas, B. Zikopoulos, H. Barbas, The Structural Model: a theory linking connections, plasticity, pathology, development and evolution of the cerebral cortex, *Brain Struct. Funct.* 224 (2019) 985–1008.
- [147] M.Á. García-Cabezas, J.L. Hacker, B. Zikopoulos, A Protocol for cortical type analysis of the human Neocortex applied on Histological samples, the Atlas of von Economo and Koskinas, and Magnetic Resonance imaging, *Front. Neuroanat.* 14 (2020) 576015.
- [148] C. Paquola, et al., Microstructural and functional gradients are increasingly dissociated in transmodal cortices, *PLoS Biol.* 17 (2019) e3000284.
- [149] H. Barbas, General cortical and special prefrontal connections: principles from structure to function, *Annu. Rev. Neurosci.* 38 (2015) 269–289.
- [150] N.T. Markov, et al., Anatomy of hierarchy: feedforward and feedback pathways in macaque visual cortex, *J. Comp. Neurol.* 522 (2014) 225–259.
- [151] S.F. Beul, Towards a “canonical” agranular cortical microcircuit, *Front. Neuroanat.* 8 (2015).
- [152] P.R. Huttenlocher, C. de Courten, L.J. Garey, H. Van der Loos, Synaptogenesis in human visual cortex — evidence for synapse elimination during normal development, *Neurosci. Lett.* 33 (1982) 247–252.
- [153] R. Peter, H. Synaptic Density in human frontal cortex — developmental changes and effects of aging, *Brain Res.* 163 (1979) 195–205.
- [154] B.D. Semple, K. Blomgren, K. Gimlin, D.M. Ferriero, L.J. Noble-Haeusslein, Brain development in rodents and humans: Identifying benchmarks of maturation and vulnerability to injury across species, *Prog. Neurobiol.* 106–107 (2013) 1–16.
- [155] R.L. Buckner, F.M. Krienen, The evolution of distributed association networks in the human brain, *Trends Cognit. Sci.* 17 (2013) 648–665.
- [156] J. Achterberg, D. Akarca, D.J. Strouse, J. Duncan, D.E. Astle, Spatially embedded recurrent neural networks reveal widespread links between structural and functional neuroscience findings, *Nat. Mach. Intell.* (2023). URL, <https://www.nature.com/articles/s42256-023-00748-9>.
- [157] O. Sporns, R.F. Betzel, Modular brain networks, *Annu. Rev. Psychol.* 67 (2016) 613–640.
- [158] D.S. Bassett, E.T. Bullmore, Small-world brain networks Revisited, *Neuroscientist* 23 (2017) 499–516.
- [159] J.L.S.M. Tanner, L. Coletta, A. Gozzi, R.F. Betzel, Functional Connectivity Modules in Recurrent Neural Networks: Function, Origin and Dynamics, 2023. URL, <https://arxiv.org/abs/2310.20601>. Publisher: arXiv Version Number: 1.
- [160] P. Fries, Neuronal gamma-band synchronization as a fundamental process in cortical computation, *Annu. Rev. Neurosci.* 32 (2009) 209–224.

- [161] A. Loebel, M. Tsodyks, Computation by ensemble synchronization in recurrent networks with synaptic depression, *J. Comput. Neurosci.* 13 (2002) 111–124.
- [162] Z. Li, Z. Duan, G. Chen, L. Huang, Consensus of multiagent systems and synchronization of complex networks: a unified viewpoint, *IEEE Transact. Circuits Syst. I: Regular Pap.* 57 (2009) 213–224.
- [163] N.F. Rulkov, M.M. Sushchik, L.S. Tsimring, H.D. Abarbanel, Generalized synchronization of chaos in directionally coupled chaotic systems, *Phys. Rev.* 51 (1995) 980.
- [164] L.M. Pecora, T.L. Carroll, Synchronization in chaotic systems, *Phys. Rev. Lett.* 64 (1990) 821.