Interrogating theoretical models of neural computation with deep inference

Sean R. Bittner[1], Agostina Palmigiano[1], Alex T. Piet[2,3,4], Chunyu A. Duan[5], Carlos D. Brody[2,3,6], Kenneth D. Miller[1], and John P. Cunningham[7].

[1]Department of Neuroscience, Columbia University,

[2]Princeton Neuroscience Institute,

[3]Princeton University,

[4]Allen Institute for Brain Science,

[5]Institute of Neuroscience, Chinese Academy of Sciences,

[6]Howard Hughes Medical Institute,

[7]Department of Statistics, Columbia University

1 Abstract

A cornerstone of theoretical neuroscience is the circuit model: a system of equations that captures a hypothesized neural mechanism. Such models are valuable when they give rise to an experimentally observed phenomenon – whether behavioral or a pattern of neural activity – and thus can offer insights into neural computation. The operation of these mechanistic circuits, like all models, critically depends on the choices of model parameters. A key process in circuit modeling is then to identify the model parameters consistent with observed phenomena: to solve the inverse problem. To solve challenging inverse problems modeling neural datasets, neuroscientists have used statistical inference techniques to much success. However, most research in theoretical neuroscience focuses on how computation emerges in biologically interpretable circuit models, and how the model parameters govern computation; it is not focused on the latent structure of empirical models of noisy experimental datasets. In this work, we present a novel technique that brings the power and versatility of the probabilistic modeling toolkit to theoretical inverse problems. Our method uses deep neural networks to learn parameter distributions with rich structure that have specific computational properties in biologically relevant models. This methodology is explained through a motivational example inferring conductance parameters in an STG subcircuit model. Then, with RNNs of increasing size, we show that only EPI allows precise control over the behavior of inferred parameters, and that EPI scales better in parameter dimension than alternative techniques. In the remainder of this work, we explain novel theoretical insights through the examination of intricate parametric structure in complex circuit models. In a model of primary visual cortex with multiple

neuron-types, where analysis becomes untenable with each additional neuron-type, we discovered how noise distributed across neuron-types governs the excitatory population. Finally, in a model of superior colliculus, we identified and characterized two distinct regimes of connectivity that facilitate switching between opposite tasks amidst interleaved trials. We also found that all task-switching connectivities in this model reproduce behaviors from inactivation experiments, further establishing this hypothesized circuit model. Beyond its scientific contribution, this work illustrates the variety of analyses possible once deep learning is harnessed towards solving theoretical inverse problems.

2 Introduction

The fundamental practice of theoretical neuroscience is to use a mathematical model to understand neural computation, whether that computation enables perception, action, or some intermediate processing. A neural circuit is systematized with a set of equations – the mechanistic model – and these equations are motivated by biophysics, neurophysiology, and other conceptual considerations [1–4]. The function of this system is governed by the choice of model *parameters*, which when configured in a particular way, give rise to a measurable signature of a computation. The work of analyzing a model then requires solving the inverse problem: given a computation of interest, how can we reason about particular parameter configurations? The inverse problem is crucial for reasoning about likely parameter values, uniquenesses and degeneracies, and predictions made by the model [5, 6].

Consider the idealized practice: one carefully designs a model and analytically derives how computational properties determine model parameters. Seminal examples of this gold standard include our field's understanding of memory capacity in associative neural networks [7], chaos and autocorrelation timescales in random neural networks [8], the paradoxical effect [9], and decision making [10]. Unfortunately, as circuit models include more biological realism, theory via analytical derivation becomes intractable. Still, we can gain insight into these complex models by identifying the distribution of parameters that produce computations. By solving the inverse problem in this way, scientific analysis of biologically realistic models is made possible [6, 11–14].

While theoretical neuroscience is concerned with how model parameters govern computational properties, existing methodology for statistical inference in neuroscience [15–36] (see review, [37]) requires that parameters be conditioned on an explicit dataset. The scientific insight for a model of computation is then limited by the quantity and quality of available neural data. Even with a vast amount of high-quality recordings, neural data often reflect uninstructed behaviors [38–40], and thus may only reflect the computation of interest amidst a sea of task-irrelevant factors. A common alternative is to synthesize an explicit dataset that is exemplary of that computation, so that the framework of statistical inference can be applied for parameter identification. In this case, well-defined computational properties are being shoehorned into artificial datasets for the purpose of methodological compatibility.

Another key challenge is that as models of computation become more complex, statistical inference becomes intractable. Such mechanistic models in theoretical neuroscience are noisy systems of differential equations that can only be sampled or realized through forward simulation [41, 42]; they lack a tractable likelihood function, which is necessary for statistical inference. Therefore, the most popular approaches to parameter inference in mechanistic models have been simulation-based inference methods [43,44], in which reasonable parameters are obtained via simulation and rejection. A new class of techniques [45–47] use deep learning to improve upon traditional simulation-based inference approaches. However, to use these methods in theoretical neuroscience, we must represent computation with an explicit dataset in some way. Theorists are therefore barred from using the probabilistic modeling toolkit for science with circuit models, unless they reformulate their inverse problem into a framework for observational datasets.

To address the methodological incongruity between explicit datasets and emergent properties, we present a statistical inference method for conditioning parameters of neural circuit models directly on computation. In this work, we define computation by an emergent property, which is a statistical description of the phenomena to be produced by the neural circuit model. In emergent property inference (EPI), we infer the distribution of model parameters that produce this emergent property. With EPI, parameters are conditioned directly on an implicit dataset defined by the computation of interest. By using recent optimization techniques [48], EPI uses deep learning to make rich, flexible approximations to the parameter distributions [49], the structure of which reveals scientific insight about how parameters govern the emergent property.

Equipped with this method, we prove out the potential of EPI by demonstrating its capabilities and presenting novel theoretical findings borne from its analysis. First, we show EPI's ability to handle mechanistic models using a classic model of parametric degeneracy in biology: the stomatogastric ganglion [50, 51]. Then, we show EPI's scalability to high dimensional parameter distributions by inferring connectivities of recurrent neural networks (RNNs) that exhibit stable, yet amplified responses – a hallmark of neural responses throughout the brain [52–54]. In a model of primary visual cortex (V1) [55, 56] with different neuron-types, we show that the equation for excitatory variability become analytically intractable as more populations are added. Strikingly, the way in which noisy inputs across neuron-types governs excitatory variability is salient in the visualized structure of the EPI inferred parameter distribution. Finally, we investigated the possible connectivities of superior colliculus (SC) that allow execution of different tasks on intervleaved trials [57]. EPI discovered a rich distribution containing two connectivity regimes with different solution classes. We queried the deep probability distribution learned by EPI to produce a mechanistic understanding of cortical responses in each regime. Intriguingly, all inferred connectivities reproduced results from optogenetic inactivation experiments in this behavioral paradigm – emergent phenomena that EPI was not conditioned upon. These theoretical insights afforded by EPI illustrate the value of deep inference for the interrogation of neural circuit models.

3 Results

3.1 Motivating emergent property inference of theoretical models

Consideration of the typical workflow of theoretical modeling clarifies the need for emergent property inference. First, one designs or chooses an existing model that, it is hypothesized, captures the computation of interest. To ground this process in a well-known example, consider the stomatogastric ganglion (STG) of crustaceans, a small neural circuit which generates multiple rhythmic muscle activation patterns for digestion [58]. Despite full knowledge of STG connectivity and a precise characterization of its rhythmic pattern generation, biophysical models of the STG have complicated relationships between circuit parameters and computation [12,50]. A subcircuit model of the STG [51] is shown schematically in Figure 1A. The jagged connections indicate electrical coupling having electrical conductance g_{el}, smooth connections in the diagram are inhibitory synaptic projections having strength g_{synA} onto the hub neuron, and $g_{\text{synB}} = 5\text{nS}$ for mutual inhibitory connections. Note that the behavior of this model will be critically dependent on its parameterization – the choices of conductance parameters $\mathbf{z} = [g_{\text{el}}, g_{\text{synA}}]$. Specifically, the two fast neurons ($f1$ and $f2$) mutually inhibit one another, and oscillate at a faster frequency than the mutually inhibiting slow neurons ($s1$ and $s2$). The hub neuron (hub) couples with either the fast or slow population, or both.

Second, once the model is selected, one must specify what the model should produce. In this STG model, we are concerned with neural spiking frequency, which emerges from the dynamics of the circuit model 1B. An emergent property studied by Guttierez et al. of this stochastic model is the hub neuron firing at an intermediate frequency between the intrinsic spiking rates of the fast and slow populations. This emergent property is shown in Figure 1C at an average frequency of 0.55Hz. Our notion of intermediate hub frequency is not strictly 0.55Hz, but also moderate deviations of this frequency between the fast (.35Hz) and and slow (.68Hz) frequencies, which are quantified in the emergent property with variance 0.025^2Hz^2.

Third, the model parameters producing these outputs are inferred. To infer the STG parameters of intermediate hub frequency with existing methodology, we need an explicit dataset: experimentally recorded or synthesized. By precisely quantifying the emergent property of interest as a statistical feature of the model, we use EPI to condition directly on this emergent property. EPI learns a probability distribution of model parameters constrained to produce the emergent property. In this last step lies the opportunity for a shift away from a dataset-oriented representation of model output towards that of an implicit dataset, where the only structure is the emergent property of interest.

Before presenting technical details (in the following section), let us understand emergent property inference schematically. EPI (Fig. 1D) takes, as input, the model and the specified emergent property, and as its output, produces the parameter distribution EPI (Fig. 1E). This distribution – represented for clarity as samples from the distribution – is a parameter distribution that produces the emergent property.

3.2 A deep generative modeling approach to emergent property inference

Emergent property inference (EPI) formalizes the three-step procedure of the previous section with deep probability distributions. First, as is typical, we consider the model as a coupled set of differential equations. In this STG example, the model activity $\mathbf{x} = [x_{\text{f1}}, x_{\text{f2}}, x_{\text{hub}}, x_{\text{s1}}, x_{\text{s2}}]$ is the membrane potential for each neuron, which evolves according to the biophysical conductance-based equation:

$$C_m \frac{d\mathbf{x}(t)}{dt} = -h(\mathbf{x}(t); \mathbf{z}) + d\mathbf{B} \tag{1}$$

where C_m=1nF, and $\mathbf{h}$ is a sum of the leak, calcium, potassium, hyperpolarization, electrical, and synaptic currents, all of which have their own complicated dependence on activity $\mathbf{x}$ and parameters $\mathbf{z} = [g_{\text{el}}, g_{\text{synA}}]$, and $d\mathbf{B}$ is white gaussian noise [51, 59] (see Section 5.2.1 for more detail).

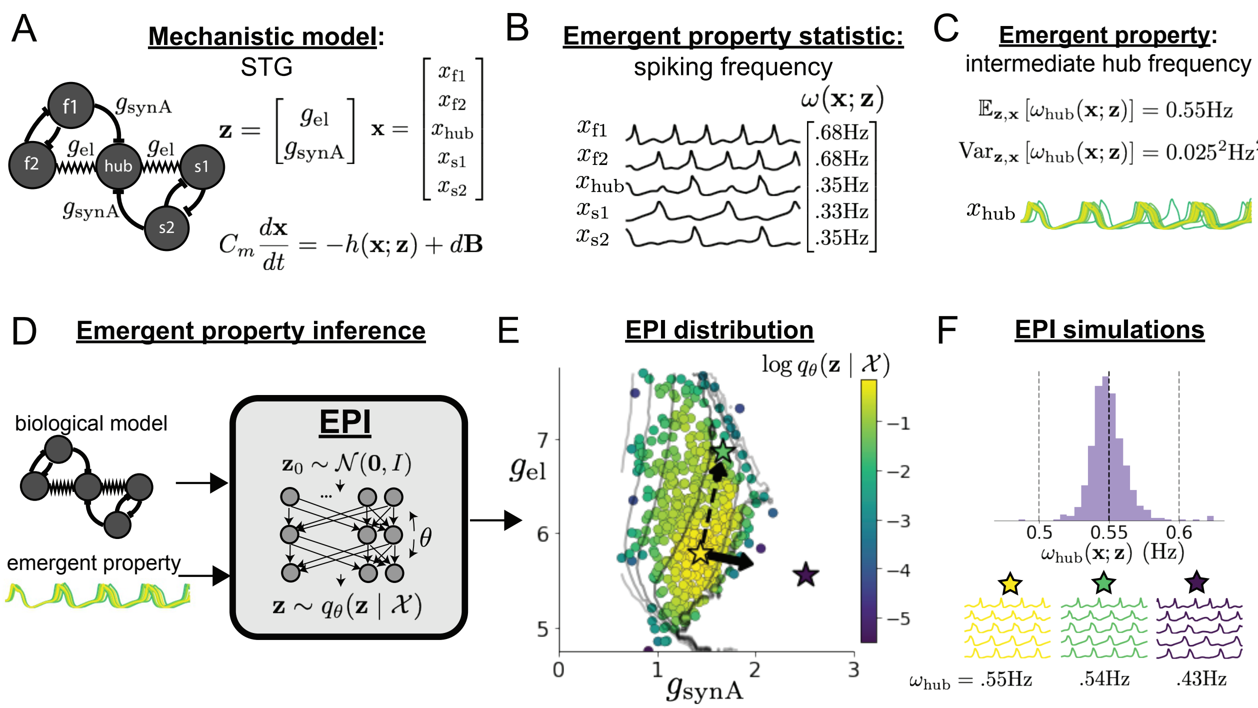

Figure 1: Emergent property inference (EPI) in the stomatogastric ganglion. **A**. Conductance-based biophysical model of the STG subcircuit. **B**. Spiking frequency $\omega(\mathbf{x};\mathbf{z})$ is an emergent property statistic. Simulated at $g_{\text{el}} = 4.5\text{nS}$ and $g_{\text{synA}} = 3\text{nS}$. **C**. The emergent property of intermediate hub frequency. Simulated activity traces are colored by $\log q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$ of generating parameters. (Panel E). **D**. For a choice of model and emergent property, emergent property inference (EPI) learns a deep probability distribution of parameters $\mathbf{z}$. **E**. The EPI distribution producing intermediate hub frequency. Samples are colored by log probability density. Contours of hub neuron frequency error are shown at levels of .525, .53, ... .575 Hz (dark to light gray away from mean). Dimension of sensitivity $\mathbf{v}_1$ (solid) and degeneracy $\mathbf{v}_2$ (dashed). **F** (Top) The predictive distribution of EPI. The black and gray dashed lines show the mean and two standard deviations according the emergent property. (Bottom) Simulations at the starred parameter values.

Second, we stipulate that our model should produce the emergent property of "intermediate hub frequency" (Figure 1C). We stipulate that the hub neuron's spiking frequency – denoted $\omega_{\text{hub}}(\mathbf{x})$ is close to a frequency of 0.55Hz, between that of the slow and fast frequencies. Mathematically, we define this emergent property with two statistical constraints: that the mean hub frequency is 0.55Hz,

$$\mathbb{E}_{\mathbf{z},\mathbf{x}}\left[\omega_{\text{hub}}(\mathbf{x};\mathbf{z})\right] \quad = \quad 0.55 \tag{2}$$

and that the variance of the hub frequency is moderate

$$\text{Var}_{\mathbf{z},\mathbf{x}}\left[\omega_{\text{hub}}(\mathbf{x};\mathbf{z})\right] \quad = \quad 0.025^2. \tag{3}$$

The hub neuron frequency is constrained over the distribution of parameters $\mathbf{z}$ and the distribution of the data $\mathbf{x}$ that those parameters produce. Formally, the emergent property is the collection of these two constraints

$$\mathcal{X}: \quad \mathbb{E}_{\mathbf{z},\mathbf{x}}\left[\omega_{\text{hub}}(\mathbf{x};\mathbf{z})\right] = 0.55, \quad \text{Var}_{\mathbf{z},\mathbf{x}}\left[\omega_{\text{hub}}(\mathbf{x};\mathbf{z})\right] = 0.025^2. \tag{4}$$

In general, an emergent property is a collection of first-, second- and higher moments of statistics that together define the phenomena.

Third, we perform emergent property inference: we find a distribution over parameter configurations $\mathbf{z}$ that produces the emergent property; in other words, they obey the constraints introduced in Equation 4. This distribution will be chosen from a family of probability distributions $\mathcal{Q} = \{q_{\boldsymbol{\theta}}(\mathbf{z}) : \boldsymbol{\theta} \in \Theta\}$, defined by a deep neural network [49, 60, 61] (Figure 1D, EPI box). Deep probability distributions map a simple random variable $\mathbf{z}_0$ through a deep neural network with weights and biases $\boldsymbol{\theta}$ to parameters $\mathbf{z} = g_{\boldsymbol{\theta}}(\mathbf{z}_0)$ to a suitably complicated distribution (see Section 5.1.2 for more details). Many distributions in $\mathcal{Q}$ will respect the emergent property constraints, so we select the most random (highest entropy) distribution, which is the same choice made in Bayesian posterior inference (see Section 5.1.6). In EPI optimization, stochastic gradient steps in $\boldsymbol{\theta}$ are taken such that entropy is maximized, and the emergent property $\mathcal{X}$ is produced (see Section 5.1) The inferred EPI distribution is denoted $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$, since it is conditioned upon emergent property $\mathcal{X}$. This is meant to share the same notation as a posterior distribution $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x})$ that is conditioned upon an explicit dataset.

EPI produces parameter distributions that can be queried for scientific insight. The modes of $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$ indicate parameter choices exemplary of the emergent property (Fig. 1E yellow star). As probability in the EPI inferred distribution decreases, the emergent property deteriorates. Perturbing $\mathbf{z}$ along a dimension in which $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$ does not change will not disturb the emergent property, making this parameter combination *degenerate* with respect to the emergent property. In contrast, if $\mathbf{z}$ is perturbed along a dimension that strongly decreases $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$, we call that parameter combination *sensitive*. By querying the second order derivative (Hessian) of $\log q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$ at a mode, we can quantitatively identify how sensitive (or robust) each eigenvector is by its eigenvalue; the more negative the eigenvalue, the more sensitive. Indeed, samples equidistant from the mode along these EPI-identified dimensions of sensitivity (v_1, smaller eigenvalue) and robustness (v_2, greater eigenvalue) (Fig. 1E, arrows) agree with error contours (Fig. 1E contours) and have diminished or preserved hub frequency, respectively (Fig. 1F activity traces). Once an EPI distribution has been inferred, this Hessian calculation requires trivial computation (when the correct architecture class is chosen, see Section 5.1.2).

In the following sections, we demonstrate EPI on three neural circuit models across ranges of biological realism, neural system function, and network scale. First, we demonstrate the superior scalability of EPI compared to alternative techniques by inferring high-dimensional distributions of recurrent neural network (RNN) connectivities that exhibit amplified, yet stable responses. Also in this RNN example, we emphasize that EPI is the only technique that controls the predictions made by the inferred parameter distribution. Next, in a model of primary visual cortex [55, 56], we show how EPI captures a curved manifold of parametric degeneracy, revealing how input variability across neuron types affects the excitatory population. Finally, in a model of superior colliculus [57], we used EPI to capture multiple parametric regimes of task switching, and queried the dimensions of sensitivity ($\mathbf{v}_1(\mathbf{z})$) to mechanistically characterize each regime.

3.3 Scaling inference of RNN connectivity with EPI

Transient amplification is a hallmark of neural activity throughout cortex, and is often thought to be intrinsically generated by recurrent connectivity in the responding cortical area [52–54]. It has been shown that to generate such amplified, yet stabilized responses, the connectivity of RNNs must be non-normal [52, 62], and satisfy additional constraints [63]. In theoretical neuroscience, RNNs are optimized and then examined to show how dynamical systems could execute a given computation [64, 65], but such biologically realistic constraints on connectivity are ignored during optimization for practical reasons. In general, access to distributions of connectivity adhering to theoretical criteria like stable amplification, chaotic fluctuations [8], or low tangling [66] would add scientific value and context to existing research with RNNs. Here, we use EPI to learn RNN connectivities producing stable amplification, and demonstrate the superior scalability and efficiency of EPI to alternative approaches.

We consider a rank-2 RNN with N neurons having connectivity $W = UV^\top$ and dynamics

$$\tau \dot{\mathbf{x}} = -\mathbf{x} + W\mathbf{x}, \tag{5}$$

where $U = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 \end{bmatrix} + g\chi^{(U)}$, $V = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{bmatrix} + g\chi^{(V)}$, $\mathbf{u}_1\mathbf{u}_2, \mathbf{v}_1, \mathbf{v}_2 \in [-1,1]^N$, and $\chi^{(U)}_{i,j}, \chi^{(V)}_{i,j} \sim \mathcal{N}(0,1)$. We infer connectivity parameterizations $\mathbf{z} = \left[\mathbf{u}_1^\top, \mathbf{u}_2^\top, \mathbf{v}_1^\top, \mathbf{v}_2^\top\right]^\top$ that produce stable amplification. Two conditions are necessary and sufficient for RNNs to exhibit stable amplification [63]: $\text{real}(\lambda_1) < 1$ and $\lambda_1^s > 1$, where λ_1 is the eigenvalue of W with greatest real part and λ^s is the maximum eigenvalue of $W^s = \frac{W+W^\top}{2}$. RNNs with $\text{real}(\lambda_1) = 0.5 \pm 0.5$ and $\lambda_1^s = 1.5 \pm 0.5$ will be stable with modest decay rate ($\text{real}(\lambda_1)$ close to its upper bound of 1) and exhibit modest amplification (λ_1^s close to its lower bound of 1). EPI can naturally condition on this emergent property

$$\begin{aligned} \mathcal{X} \quad : \quad \mathbb{E}_{\mathbf{z},\mathbf{x}} \begin{bmatrix} \text{real}(\lambda_1) \\ \lambda_1^s \end{bmatrix} &= \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix} \\ \text{Var}_{\mathbf{z},\mathbf{x}} \begin{bmatrix} \text{real}(\lambda_1) \\ \lambda_1^s \end{bmatrix} &= \begin{bmatrix} 0.25^2 \\ 0.25^2 \end{bmatrix}, \end{aligned} \tag{6}$$

under the notion that variance constraints with standard deviation 0.25 predicate that the vast majority of samples (those within two standard deviations) are within the specified ranges.

For comparison, we infer the parameters $\mathbf{z}$ likely to produce stable amplification using two alternative simulation-based inference approaches. We ran sequential Monte Carlo approximate Bayesian computation (SMC-ABC) [43] and sequential neural posterior estimation (SNPE) [45] with observation $\mathbf{x}_0 = \boldsymbol{\mu}$. SMC-ABC is a rejection sampling approach that SMC techniques to improve efficiency, and SNPE approximates posteriors with deep probability distributions using a two-network architecture (see Section 5.1.1). Unlike EPI, these statistical inference techniques do not constrain the statistics of the predictive distribution, and these predictions of the inferred posteriors are typically affected by model characteristics (e.g. N and g, Fig. 11). To compare the efficiency of these different techniques, we measured the time and number of simulations necessary for the distance of the predictive mean to be less than 0.5 from $\boldsymbol{\mu} = \mathbf{x}_0$ (see Section 5.2.2).

As the number of neurons N in the RNN is scaled, and thus the dimension of the parameter space $\mathbf{z} \in [-1,1]^{4N}$, we see that EPI converges at greater speed and at greater dimension than SMC-ABC and SNPE (Fig. 2A). It also becomes most efficient to use EPI in terms of simulation count at $N = 50$ (Fig. 2B). It is well known that ABC techniques struggle in dimensions greater than about 30 [67], yet we were careful to assess the scalability of the more comparable approach

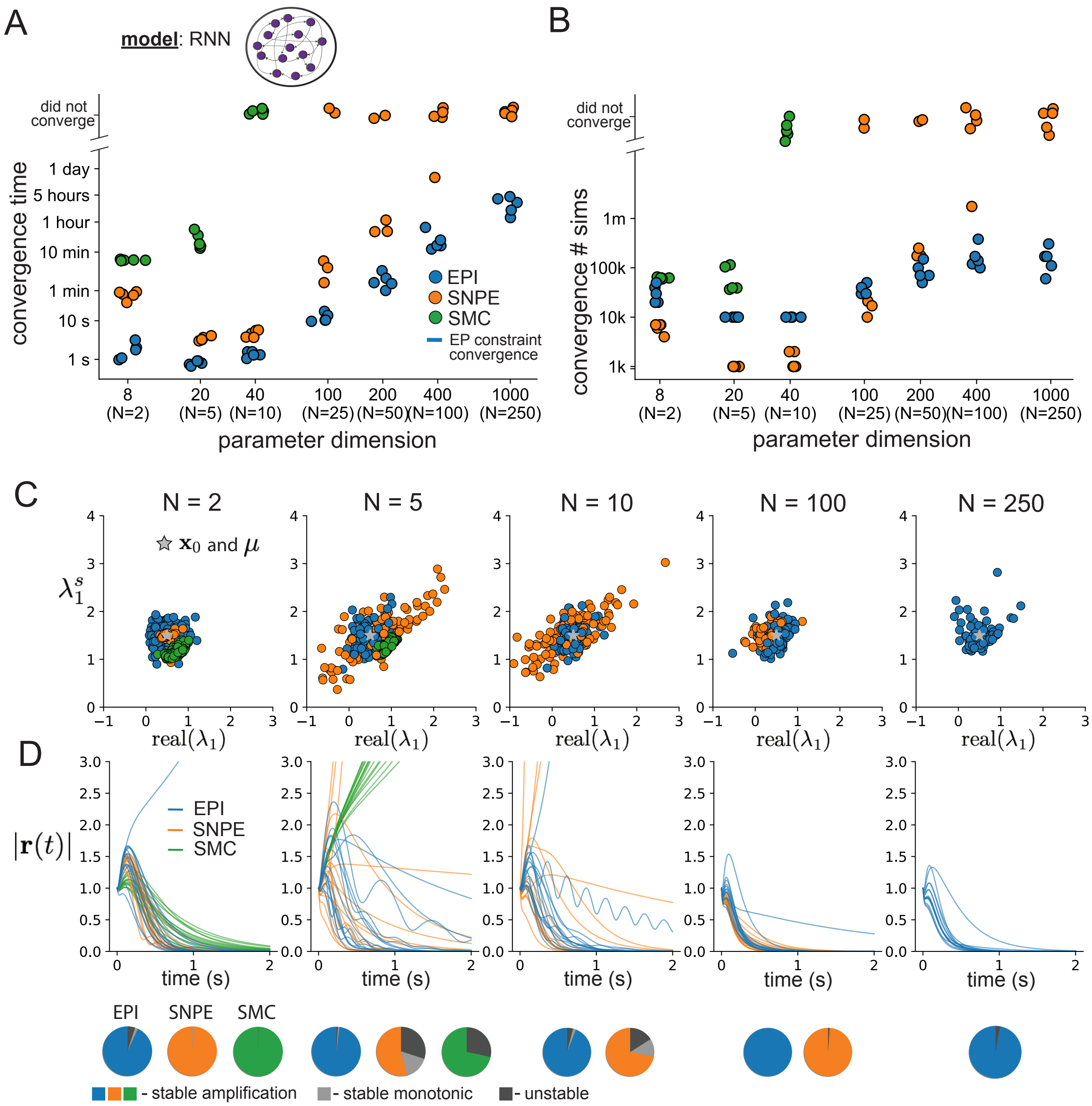

Figure 2: **A**. Wall time of EPI (blue), SNPE (orange), and SMC-ABC (green) to converge on RNN connectivities producing stable amplification. Each dot shows convergence time for an individual random seed. For reference, the mean wall time for EPI to achieve its full constraint convergence (means and variances) is shown (blue line). **B**. Simulation count of each algorithm to achieve convergence. Same conventions as A. **C**. The predictive distributions of connectivities inferred by EPI (blue), SNPE (orange), and SMC-ABC (green), with reference to $\mathbf{x}_0 = \boldsymbol{\mu}$ (gray star). **D**. Simulations of networks inferred by each method ($\tau = 100ms$). Each trace (15 per algorithm) corresponds to simulation of one z. (Below) Ratio of obtained samples producing stable amplification, monotonic decay, and instability.

SNPE. Between EPI and SNPE, we closely controlled the number of parameters in deep probability distributions by dimensionality (Fig. 10), and tested more aggressive SNPE hyperparameterizations when SNPE failed to converge (Fig. 12). From this analysis, we see that deep inference techniques EPI and SNPE are far more amenable to inference of high dimensional parameter distributions than rejection sampling techniques like SMC-ABC, and that EPI outperforms SNPE in both criteria in high dimensions.

No matter the number of neurons, EPI always produces connectivity distributions with mean and variance of real(λ_1) and λ_1^s according to $\mathcal{X}$ (Fig. 2C, blue). For the dimensionalities in which SMC-ABC is tractable, the inferred parameters are concentrated and offset from $\mathbf{x}_0$ (Fig. 2C, green). When using SNPE the predictions of the inferred parameters are highly concentrated at some RNN sizes and widely varied in others (Fig. 2C, orange). We see these properties reflected in simulations from the inferred distributions: EPI produces a consistent variety of stable, amplified activity norms $|r(t)|$, SMC-ABC produces a limited variety of responses, and the changing variety of responses from SNPE emphasizes the control of EPI on parameter predictions.

EPI outperforms SNPE in high dimensions by using gradient information (from $\nabla_{\mathbf{z}} f(\mathbf{x}; \mathbf{z}) = \nabla_{\mathbf{z}} [\text{real}(\lambda_1), \lambda_1^s]^\top$) on each iteration of optimization. This agrees with recent speculation that such gradient information could improve the efficiency of simulation-based inference techniques [68]. Since gradients of the emergent property statistics are necessary in EPI optimization, gradient tractabillity is a key criteria when determining the suitability of a simulation-based inference technique. Evidenced by this analysis, EPI is a clear choice for inferring high dimensional parameter distributions when the emergent property gradient is efficiently calculated. This can be invaluable for understanding how RNNs produce complex emergent phenomena. Even with a high degree of biophysical realism and expensive emergent property gradients, EPI was run successfully on intermediate hub frequency in a 5-neuron subcircuit model of the STG (Section 3.1). However, conditioning on the pyloric rhythm [69] in a model of the pyloric subnetwork model [12] proved to be prohibitive with EPI. The pyloric subnetwork requires many time steps for simulation and many key emergent property statistics (e.g. burst duration and phase gap) are not easily calculated or approximated with differentiable functions. In such cases, gradient-free approaches like SNPE have proved to be a powerful option [45]. In the next two sections, we use EPI for novel scientific insight by examining the structure of inferred distributions.

3.4 EPI reveals how noisy input across neuron-types governs excitatory variability in a V1 model

Dynamical models of excitatory (E) and inhibitory (I) populations with supralinear input-output function have succeeded in explaining a host of experimentally documented phenomena. In a regime characterized by inhibitory stabilization of strong recurrent excitation, these models give rise to paradoxical responses [9], selective amplification [52,62], surround suppression [70] and normalization [71]. Despite their strong predictive power, E-I circuit models rely on the assumption that inhibition can be studied as an indivisible unit. However, experimental evidence shows that inhibition is composed of distinct elements – parvalbumin (P), somatostatin (S), VIP (V) – composing 80% of GABAergic interneurons in V1 [72–74], and that these inhibitory cell types follow specific connectivity patterns (Fig. 3A) [75]. While research has shown that V1 only shares specific dimensions of neuronal variability with downstream areas [76], the role played by recurrent dynamics and the connectivity across neuron-type populations is not understood. Here, in a model of V1 with biologically realistic connectivity, we use EPI to show how the structure of input across neuron types affects the variability of the excitatory population – the population largely responsible for projecting to other brain areas [77].

We considered response variability of a nonlinear dynamical V1 circuit model (Fig. 3A) with a state comprised of each neuron-type population's rate $\mathbf{x} = [x_E, x_P, x_S, x_V]^\top$. Each population receives recurrent input $W\mathbf{x}$, where W is the effective connectivity estimated from post-synaptic potential and connectivity rate measurements (see Section 5.2.3). Each population also experiences an external input $\mathbf{h}$, which determines population rate via supralinear nonlinearity $\phi(\cdot) = [\cdot]_+^2$. To build on previous work, we model visual contrast-dependent input to the E- and P-populations $\mathbf{h} = \mathbf{b} + c\mathbf{h}_c$. There is also an additive noisy input $\boldsymbol{\epsilon}$ parameterized by variances for each neuron type population $\mathbf{z} = \boldsymbol{\sigma}^2 = \left[\sigma_E^2, \sigma_P^2, \sigma_S^2, \sigma_V^2\right]$. This noise has a slower dynamical timescale $\tau_{\text{noise}} > \tau$ then the population rate, allowing fluctuations around a stimulus-dependent steady-state

$$\tau \frac{d\mathbf{x}}{dt} = -\mathbf{x} + \phi(W\mathbf{x} + \mathbf{h} + \boldsymbol{\epsilon}). \tag{7}$$

This model is the stochastic stabilized supralinear network (SSSN) [78] generalized to have multiple inhibitory neuron types, and introduces stochasticity to previous four neuron-type models of V1 [55]. Stochasticity and inhibitory multiplicity introduce substantial complexity to mathematical derivations (see Section 5.2.4) motivating the treatment of this model with EPI. Here, we consider fixed weights W and input $\mathbf{h}$ [56] (Fig. 3B), and study the effect of input variability

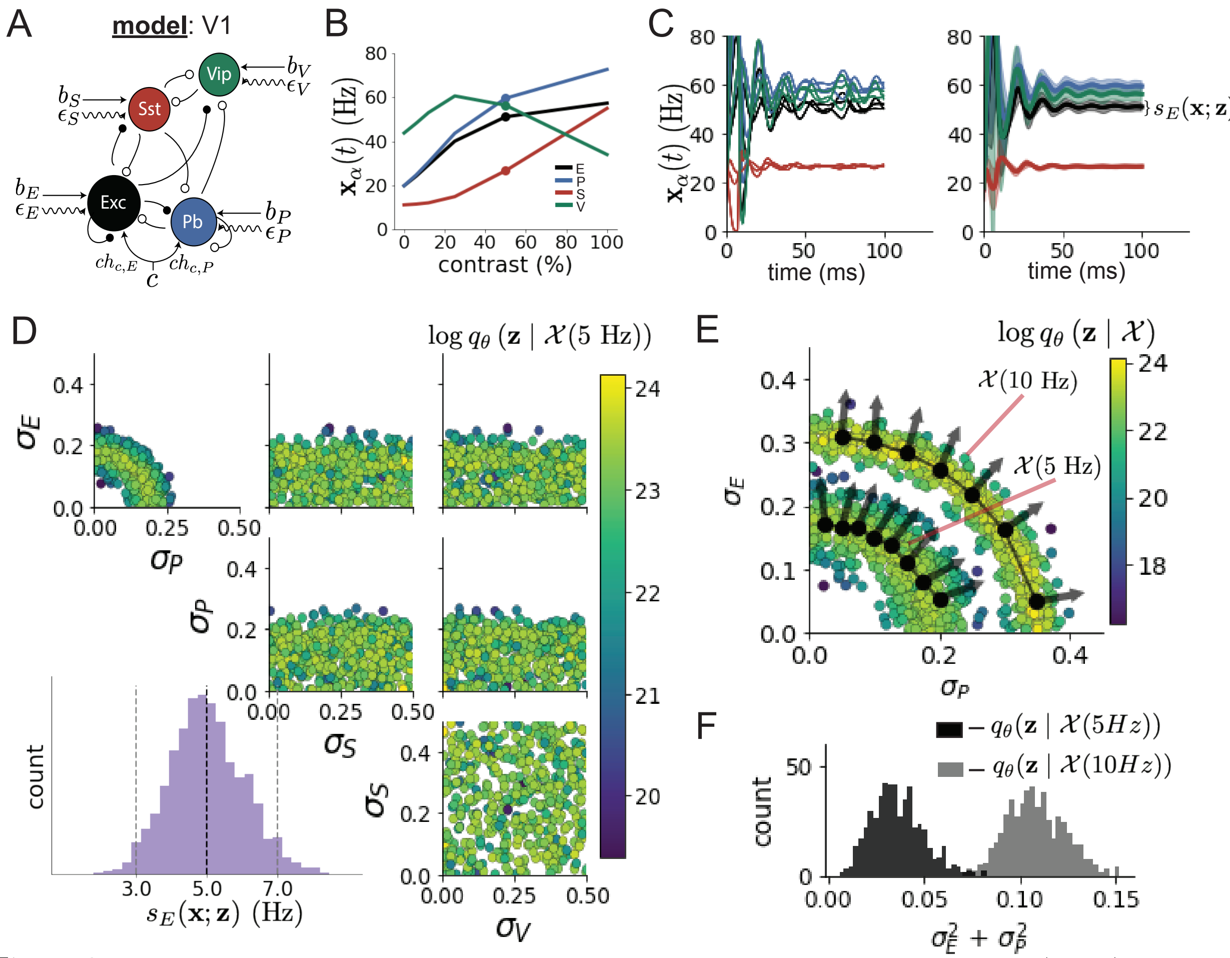

Figure 3: Emergent property inference in the stochastic stabilized supralinear network (SSSN) **A**. Four-population model of primary visual cortex with excitatory (black), parvalbumin (blue), somatostatin (red), and VIP (green) neurons (excitatory and inhibitory projections filled and unfilled, respectively). Some neuron-types largely do not form synaptic projections to others ($|W_{\alpha_1,\alpha_2})| < 0.025$). Each neural population receives a baseline input $\mathbf{h}_b$, and the E- and P-populations also receive a contrast-dependent input $\mathbf{h}_c$. Additionally, each neural population receives a slow noisy input $\boldsymbol{\epsilon}$. **B**. Steady-state responses of the SSN model (deterministic, $\boldsymbol{\sigma} = \mathbf{0}$) to varying contrasts. The response at 50% contrast (dots) is the focus of our analysis. **C**. Transient network responses of the SSSN model at 50 % contrast. (Left) Traces are independent trials with varying initialization $\mathbf{x}(0)$ and noise realization. (Right) Mean (solid line) and standard deviation (shading) of responses. **D**. EPI distribution of noise parameters $\mathbf{z}$ conditioned on E-population variability. The EPI predictive distribution of $s_E(\mathbf{x};\mathbf{z})$ is show on the bottom-left. **E**. (Top) Enlarged visualization of the σ_E-σ_P marginal distribution of EPI $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}(5\text{ Hz})$ and $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}(10\text{ Hz})$. Each black dot shows the mode at each σ_P. The arrows show the most sensitive dimensions of the Hessian evaluated at these modes. **F**. The predictive distributions of $\sigma_E^2 + \sigma_P^2$ of each parameter distribution $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}(5\text{ Hz})$ and $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}(10\text{ Hz})$.

$\mathbf{z} = [\sigma_E, \sigma_P, \sigma_S, \sigma_V]^\top$ on excitatory variability at 50% contrast.

We quantify different levels y of E-population variability with the emergent property

$$\begin{aligned} \mathcal{X}(y) \quad : \quad \mathbb{E}_{\mathbf{z}}\left[s_E(\mathbf{x};\mathbf{z})\right] &= y \\ \mathrm{Var}_{\mathbf{z}}\left[s_E(\mathbf{x};\mathbf{z})\right] &= 1\mathrm{Hz}^2, \end{aligned} \tag{8}$$

where $s_E(\mathbf{x};\mathbf{z})$ is the standard deviation of the stochastic E-population response about its steady state (Fig. 3C).

We ran EPI to obtain parameter distribution $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}(5 \text{ Hz}))$ producing E-population variability around 5 Hz (Fig. 3D). From the marginal distribution of σ_E and σ_P (Fig. 3D, top-left), we can see that $s_E(\mathbf{x};\mathbf{z})$ is sensitive to various combinations of σ_E and σ_P. Alternatively, both σ_S and σ_V are degenerate with respect to $s_E(\mathbf{x};\mathbf{z})$ evidenced by the high variability in those dimensions (Fig. 3D, bottom-right). Together, these observations imply a curved path of parametric degeneracy with respect to $s_E(\mathbf{x};\mathbf{z})$ of 5 Hz, which is indicated by the modes along σ_P (Fig. 3E). The dimensions of sensitivity conferred by EPI and this plain visual structure suggest a quadratic relationship in the emergent property statistic $s_E(\mathbf{x};\mathbf{z})$ and parameters $\mathbf{z}$, which is preserved at a greater level of variability $\mathcal{X}(10 \text{ Hz})$ (Fig. 3E). Indeed, the sum of squares of σ_E and σ_P is larger in $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}(10 \text{ Hz}))$ than $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}(5 \text{ Hz}))$ (Fig 3F, $p < 1 \times 10{-}10$), while the sum of squares of σ_S and σ_V are not significantly different in the two EPI distributions (Fig. 15, $p = .40$). The strong compensatory influence of the E- and P-population input variability on excitatory varability is intriguing, since this circuit exhibited a paradoxical effect in the P-population (and no other inhibitory types) at 50% contrast (Fig. 15) meaning that the E-population is P-stabilized. Future research may uncover a link between the populations of stabilizations and compensatory interactions governing excitatory variability.

EPI revealed the quadratic relationship between $s_E(\mathbf{x};\mathbf{z})$ and $\mathbf{z}$. While this property is ultimately derivable, we show that with each additional neuron-type population, the formula becomes quite unruly and likely escapes comprehensible analysis in our case (see Section 5.2.4). This emphasizes the need for streamlined methods for gaining understanding about theoretical models when mathematical analysis becomes prohibitive.

3.5 EPI identifies two regimes of rapid task switching

It has been shown that rats can learn to switch from one behavioral task to the next on randomly interleaved trials [79], and an important question is what types of neural connectivity allow this ability. In this experimental setup, rats were explicitly cued on each trial to either orient towards a visual stimulus in the Pro (P) task or orient away from a visual stimulus in the Anti (A) task (Fig. 4A). Neural recordings in superior colliculus (SC) exhibited two populations of neurons that represented task context (Pro or Anti). Furthermore, Pro/Anti neurons in each hemisphere were strongly correlated with the animal's decision [57]. These results motivated a model of SC that is a four-population dynamical system with functionally-defined neuron-types. Here, our goal is to understand how connectivity in this circuit model governs the ability to switch tasks rapidly.

In this SC model, there are Pro- and Anti-populations in each hemisphere (left (L) and right (R)) with activity variables $\mathbf{x} = [x_{LP}, x_{LA}, x_{RP}, x_{RA}]^\top$. The connectivity of these populations is parameterized by self sW, vertical vW, diagonal dW and horizontal hW connections (Fig. 4B). The input $\mathbf{h}$ is comprised of a positive cue-dependent signal to the Pro or Anti populations, a positive stimulus-dependent input to either the Left or Right populations, and a choice-period input to the entire network (see Section 5.2.5). Model responses are bounded from 0 to 1 as a function ϕ of an internal variable $\mathbf{u}$

$$\begin{aligned} \tau \frac{d\mathbf{u}}{dt} &= -\mathbf{u} + W\mathbf{x} + \mathbf{h} + d\mathbf{B} \\ \mathbf{x} &= \phi(\mathbf{u}). \end{aligned} \tag{9}$$

The model responds to the side with greater Pro neuron activation; e.g. the response is left if $x_{LP} > x_{RP}$ at the end of the trial. Here, we use EPI to determine the network connectivity $\mathbf{z} = [sW, vW, dW, hW]^\top$ that produces rapid task switching.

We define the computation of rapid task switching as accurate execution of each task. Inferred models should not exhibit fully random responses (50%), or perfect performance (100%), since perfection is never attained by even the best trained rats. We formulate rapid task switching as an emergent property by stipulating that the average accuracy in the Pro task $p_P(\mathbf{x};\mathbf{z})$ and Anti task $p_A(\mathbf{x};\mathbf{z})$ be 75% with variance $7.5\%^2$.

$$\begin{aligned} \mathcal{X} \ : \ \mathbb{E}_{\mathbf{z}} \begin{bmatrix} p_P(\mathbf{x};\mathbf{z}) \\ p_A(\mathbf{x};\mathbf{z}) \end{bmatrix} &= \begin{bmatrix} 75\% \\ 75\% \end{bmatrix} \\ \mathrm{Var}_{\mathbf{z}} \begin{bmatrix} p_P(\mathbf{x};\mathbf{z}) \\ p_A(\mathbf{x};\mathbf{z}) \end{bmatrix} &= \begin{bmatrix} 7.5\%^2 \\ 7.5\%^2 \end{bmatrix} \end{aligned} \tag{10}$$

The EPI inferred distribution (Fig. 4C) produces task accuracies (Fig. 4C, middle-left) according to our mathematical definition of rapid task switching (Equation 10). The patterns of connectivity that govern each task accuracy are nonlinear (Fig. 17A-B); they are not captured well by linear

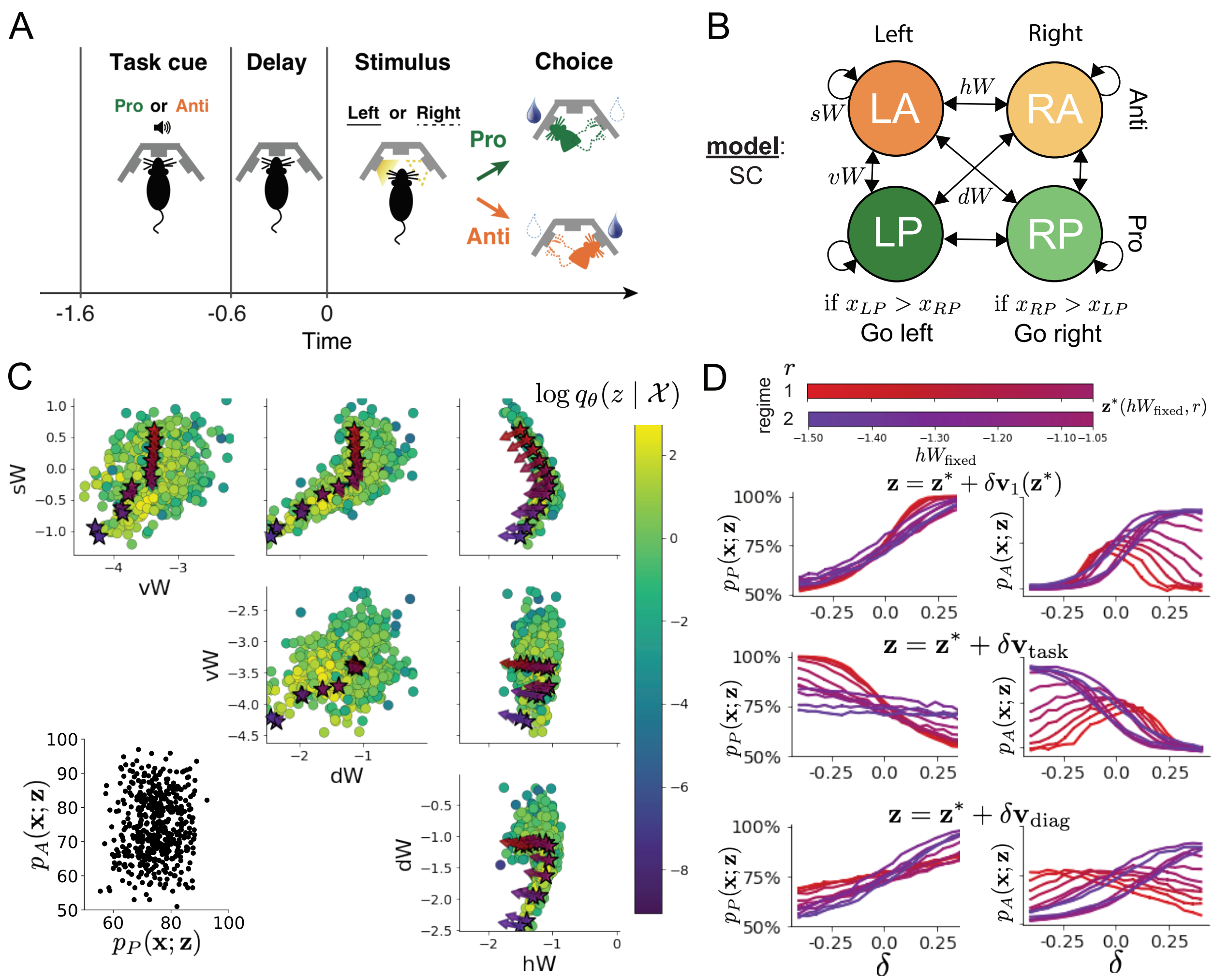

Figure 4: **A**. Rapid task switching behavioral paradigm (see text). **B**. Model of superior colliculus (SC). Neurons: LP - left pro, RP - right pro, LA - left anti, RA - right anti. Parameters: sW - self, hW - horizontal, vW -vertical, dW - diagonal weights. **C**. The EPI inferred distribution of rapid task switching networks. Red and purple stars indicate modes $\mathbf{z}^*$ of each connectivity regime. Sensitivity vectors $\mathbf{v}_1(\mathbf{z}^*)$ are shown by arrows. (Bottom-left) EPI predictive distribution of task accuracies. **D**. The connectivity regimes have different responses to perturbation. (Top) Mean and standard error ($N_{\text{test}} = 25$) of accuracy with respect to perturbation along the sensitivity dimension of each mode $\mathbf{z}^*$. (Middle) Same with perturbation in the dimension of increasing λ_{task} ($\mathbf{v}_{\text{task}}$). (Bottom) Same with perturbation in the dimension of increasing λ_{diag} ($\mathbf{v}_{\text{diag}}$).

correlations (Fig. 17C). For example, there appear to be two regimes of connectivity: the local structure of the EPI distribution changes dramatically after crossing a threshold of sW (Fig. 17A sW-hW marginal distribution). Not only has EPI captured this intricate, nonlinear distribution, we can use the distribution $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$ returned by EPI to understand these two parametric regimes of SC connectivity.

To distinguish these two parts of the distribution, we point out that for many fixed values of parameter hW, there are two modes in the EPI distribution. Thus, by fixing hW to different values and doing gradient ascent on $\log q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$ in the parameter spaces proximal to each mode, we can identify a set of modes $\mathbf{z}^*(hW_{\text{fixed}}, r)$ for each putative regime $r \in \{1, 2\}$ (see Section 5.2.5). As hW_{fixed} increases, the modes coalesce to intermediate parameters reflecting a transition between the two sets of modes (Fig. 20 top). However, the sensitivity dimensions of these modes $\mathbf{v}_1$ (refer to Section 3.2), which reflect the structure of the EPI distribution around each mode, are different across putative regime, yet consistent across hW_{fixed}. This categorical difference in sensitivity dimension across the two sets of modes shows that they indeed represent two different regimes of computation in which connectivity governs computation in different ways.

To understand how SC connectivity governs computation in each regime, we can examine how perturbations along $\mathbf{v}_1(\mathbf{z}^*)$ affect task performance in each regime; we measure task accuracy for connectivity changes in the dimension that rapid task switching is most sensitive. While the monotonic increase in Pro accuracy with $\mathbf{v}_1$ perturbation is largely unaffected by regime (Fig. 4D, top-left), we see a stark difference in Anti accuracy: Anti accuracy falls in either direction of $\mathbf{v}_1$ in regime 1, yet monotonically increases along with Pro accuracy in regime 2 (Fig. 4D, top-right). These two rapid task switching pathologies are caused by distinct connectivity changes ($\mathbf{v}_1(\mathbf{z}^*(\cdot, 1))$ vs $\mathbf{v}_1(\mathbf{z}^*(\cdot, 2))$) and explain the sharp change in local structure of the EPI distribution.

To understand the connectivity mechanisms that distinguish these two regimes, we perturb connectivity at each mode in dimensions that have well defined roles in processing for the Pro and Anti tasks. A convenient property of this connectivity parameterization is that there are $\mathbf{z}$-invariant eigenmodes of connectivity, whose eigenvalues (or degree of amplification) change with $\mathbf{z}$. These eigenmodes have intuitive roles in processing in each task, and are accordingly named the *all*, *side*, *task*, and *diag* eigenmodes (see Section 5.2.5). Furthermore, the parameter dimension $\mathbf{v}_a$ ($a \in \{\text{all}, \text{side}, \text{task}, \text{ and diag}\}$) that increases the eigenvalue of connectivity λ_a is $\mathbf{z}$-invariant (unlike the sensitivity dimension $\mathbf{v}_1(\mathbf{z})$) and $\mathbf{v}_a \perp \mathbf{v}_{b \neq a}$. Thus, by changing the degree of amplification of each processing mode by perturbing $\mathbf{z}$ along $\mathbf{v}_a$, we can elicit the differentiating properties of the two regimes.

Through these connectivity perturbation analyses, we found that increasing λ_{task} strongly reduced Pro accuracy in regime 1, yet strongly reduced Anti accuracy in regime 2. This suggests that stronger task representations can inhibit both Pro and Anti task performance in different contexts. Furthermore, changing λ_{task} in either direction decreases Anti performance in regime 1, showing that Anti task performance in regime 1 is dependent on a specific level of task representation. We also found that with increasing λ_{diag}, Pro accuracy increased in both regimes, but there were opposite effects on Anti accuracy. In regime 1, stronger amplification of diagonal population patterns decreased Anti accuracy, while in regime 2 accuracy increased. These findings give us an understanding of the mechanistic differences in computation enabling rapid task switching in each regime.

3.6 EPI inferred SC connectivities reproduce results from optogenetic inactivation experiments

During the delay period of this task, the circuit must prepare to execute the correct task based on the cue input. Experimental results from Duan et al. found that optogenetic inactivation of SC during the delay period consistently decreased performance in the Anti task, but had no effect on the Pro task (Fig. 5A). All network connectivities inferred by EPI exhibited this same effect, when network activities were silenced during the delay period (see Section 5.2.5). Notably, EPI inferred connectivities were only conditioned upon the emergent property of rapid task switching, not on Anti task failure during delay period silencing.

Similarities across Pro and Anti trials in choice period responses following delay period inactivation (Fig. 21A) suggested that connectivity patterns inducing greater Pro task accuracy increase error in delay period inactivated Anti trials (Fig. 5B). The strong anticorrelation between p_P and $p_{A,opto}$ across EPI inferred connectivities led to the following hypothesis about each connectivity regime: the sensitivity dimension of each regime decreases $p_{A,opto}$ irrespective of its effect on p_A, since both $\mathbf{v}_1$ and $\mathbf{v}_2$ increase p_P. Indeed, in regimes 1 and 2 where sensitivity dimensions elicit different responses in p_A, $p_{A,opto}$ decreases since the connectivity changes enhancing p_P exacerbate Anti trial error (Fig. 5C). Thus, the altered state caused by delay period silencing makes the connectivity governing p_P more influential on Anti accuracy than the connectivity governing p_A.

In summary, we used EPI to obtain the full distribution of connectivities that execute rapid task

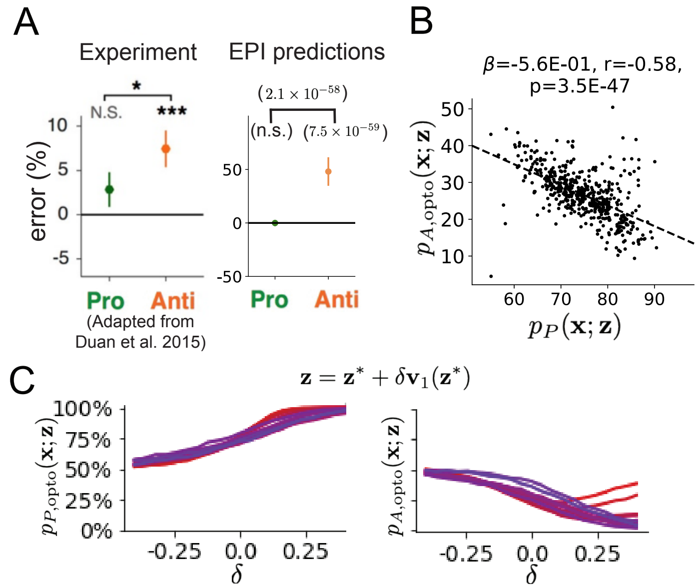

Figure 5: **A**. The EPI distribution predicts experimental results (left) showing no change in the Pro task, but larger error in the Anti task (right). **B**. Accuracy in the Anti task during delay period optogenetic inactivation $p_{A,\text{opto}}$ is strongly anticorrelated with accuracy in the Pro task. **C**. Mean and standard error ($N_{\text{test}} = 25$) of accuracy with respect to perturbation along the sensitivity dimension of each mode $\mathbf{z}^*$.

switching. This EPI distribution revealed two regimes of rapid task switching, which we characterized using the probabilistic toolkit EPI seemlessly affords. We found that both of these parametric regimes identified by EPI reproduce results from optogenetic inactivation experiments: when activity is silenced during the delay period, only Anti accuracy suffers. We then identified the connectivity mechanisms governing Anti accuracy during delay period silencing, and showed that they are regime invariant.

4 Discussion

In neuroscience, machine learning has primarily been used to reveal structure in neural datasets [37]. Careful inference procedures are developed for these statistical models allowing precise, quantitative reasoning, which clarifies the way data informs beliefs about the model parameters. However, these statistical models often lack resemblance to the underlying biology, making it unclear how to go from the structure revealed by these methods, to the neural mechanisms giving rise to it. In contrast, theoretical neuroscience has focused on careful mechanistic modeling and the production of emergent properties of computation, rather than measuring structure in some noisy observed dataset. The careful steps of *i.*) model design and *ii.*) emergent property definition, are followed by *iii.*) practical inference methods resulting in an opaque characterization of the way model parameters govern computation. In this work, we improve upon parameter inference techniques in theoretical neuroscience with emergent property inference, harnessing deep learning towards parameter inference with respect to emergent phenomena in interpretable models of neural computation (see Section 5.1.1).

Methodology for statistical inference in mechanistic models of neural circuits has evolved considerably in recent years. Early work used rejection sampling techniques [43, 80, 81], but more recently developed methodology employs deep learning to improve efficiency or provide flexible distribution approximations. SNPE [45] and other sequential techniques for inference in mechanistic models developed along with EPI (see Section 5.1.1) have been used for posterior inference with noisy experimental datasets. On the other hand, EPI is a deep inference technique designed to condition upon mathematical criteria, such that the parameter distribution only produces the specified *emergent properties* of computation. EPI is thus ideally suited for questions in theoretical neuroscience, and we show that it has superior scaling properties to these other inference techniques (see Section 3.3).

In this work, we prove out the utility of deep probability distributions for theoretical neuroscience. While previous work has used SNPE to obtain flexible posterior approximations in mechanistic models conditioned on experimental datasets, we use the rich structure captured by deep probability distributions in EPI to gain new theoretical insights. This is first done in a complex model of V1, where we combine the modeling advancements [55, 78], which make analytic characterization of excitatory variability very complicated. There, EPI clearly and simply revealed the parametric structure of input variability across neuron-type populations that governed excitatory variability, which has implications on the dimensionality and nature information transmission in visual cortex.

Finally, we uesd EPI to identify two distinct regimes of SC connectivity that enabled rapid task switching. By systematically characterizing the local structure of the inferred distribution using the analytic capabilities deep probability distributions, we discerned a mechanistic understanding of each computational regime. Each of these regimes reproduced effects from optogenetic experiments [79], suggesting that both are biologically plausible. These analyses of the V1 and SC models serve as examples of how to leverage the probabilistic toolkit for theoretical insight into models of neural computation.

Acknowledgements:

This work was funded by NSF Graduate Research Fellowship, DGE-1644869, McKnight Endowment Fund, NIH NINDS 5R01NS100066, Simons Foundation 542963, NSF NeuroNex Award, DBI-1707398, The Gatsby Charitable Foundation, Simons Collaboration on the Global Brain Postdoctoral Fellowship, Chinese Postdoctoral Science Foundation, and International Exchange Program Fellowship. Helpful conversations were had with Francesca Mastrogiuseppe, Srdjan Ostojic, James Fitzgerald, Stephen Baccus, Dhruva Raman, Liam Paninski, and Larry Abbott.

Data availability statement:

The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Code availability statement:

All software written for the current study is available at https://github.com/cunningham-lab/epi.

References

[1] Nancy Kopell and G Bard Ermentrout. Coupled oscillators and the design of central pattern generators. *Mathematical biosciences*, 90(1-2):87–109, 1988.

[2] Eve Marder. From biophysics to models of network function. *Annual review of neuroscience*, 21(1):25–45, 1998.

[3] Larry F Abbott. Theoretical neuroscience rising. *Neuron*, 60(3):489–495, 2008.

[4] Xiao-Jing Wang. Neurophysiological and computational principles of cortical rhythms in cognition. *Physiological reviews*, 90(3):1195–1268, 2010.

[5] Ryan N Gutenkunst, Joshua J Waterfall, Fergal P Casey, Kevin S Brown, Christopher R Myers, and James P Sethna. Universally sloppy parameter sensitivities in systems biology models. *PLoS Comput Biol*, 3(10):e189, 2007.

[6] Timothy O'Leary, Alex H Williams, Alessio Franci, and Eve Marder. Cell types, network homeostasis, and pathological compensation from a biologically plausible ion channel expression model. *Neuron*, 82(4):809–821, 2014.

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

[8] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural networks. *Physical review letters*, 61(3):259, 1988.

[9] Misha V Tsodyks, William E Skaggs, Terrence J Sejnowski, and Bruce L McNaughton. Paradoxical effects of external modulation of inhibitory interneurons. *Journal of neuroscience*, 17(11):4382–4388, 1997.

[10] Kong-Fatt Wong and Xiao-Jing Wang. A recurrent network mechanism of time integration in perceptual decisions. *Journal of Neuroscience*, 26(4):1314–1328, 2006.

[11] WR Foster, LH Ungar, and JS Schwaber. Significance of conductances in hodgkin-huxley models. *Journal of neurophysiology*, 70(6):2502–2518, 1993.

[12] Astrid A Prinz, Dirk Bucher, and Eve Marder. Similar network activity from disparate circuit parameters. *Nature neuroscience*, 7(12):1345–1352, 2004.

[13] Pablo Achard and Erik De Schutter. Complex parameter landscape for a complex neuron model. *PLoS computational biology*, 2(7):e94, 2006.

[14] Leandro M Alonso and Eve Marder. Visualization of currents in neural models with similar behavior and different conductance densities. *Elife*, 8:e42722, 2019.

[15] Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*, 13(8):1713–1720, 2001.

[16] Emery N Brown, Loren M Frank, Dengda Tang, Michael C Quirk, and Matthew A Wilson. A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–7425, 1998.

[17] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.

[18] Wilson Truccolo, Uri T Eden, Matthew R Fellows, John P Donoghue, and Emery N Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089, 2005.

[19] Elad Schneidman, Michael J Berry, Ronen Segev, and William Bialek. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007–1012, 2006.

[20] Shaul Druckmann, Yoav Banitt, Albert A Gidon, Felix Schürmann, Henry Markram, and Idan Segev. A novel multiple objective optimization framework for constraining conductance-based neuron models by experimental data. *Frontiers in neuroscience*, 1:1, 2007.

[21] Richard Turner and Maneesh Sahani. A maximum-likelihood interpretation for slow feature analysis. *Neural computation*, 19(4):1022–1038, 2007.

[22] M Yu Byron, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In *Advances in neural information processing systems*, pages 1881–1888, 2009.

[23] Jakob H Macke, Lars Buesing, John P Cunningham, Byron M Yu, Krishna V Shenoy, and Maneesh Sahani. Empirical models of spiking in neural populations. *Advances in neural information processing systems*, 24:1350–1358, 2011.

[24] Il Memming Park and Jonathan W Pillow. Bayesian spike-triggered covariance analysis. In *Advances in neural information processing systems*, pages 1692–1700, 2011.

[25] Einat Granot-Atedgi, Gašper Tkačik, Ronen Segev, and Elad Schneidman. Stimulus-dependent maximum entropy models of neural population codes. *PLoS Comput Biol*, 9(3):e1002922, 2013.

[26] Kenneth W Latimer, Jacob L Yates, Miriam LR Meister, Alexander C Huk, and Jonathan W Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making. *Science*, 349(6244):184–187, 2015.

[27] Kaushik J Lakshminarasimhan, Marina Petsalis, Hyeshin Park, Gregory C DeAngelis, Xaq Pitkow, and Dora E Angelaki. A dynamic bayesian observer model reveals origins of bias in visual path integration. *Neuron*, 99(1):194–206, 2018.

[28] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable continuous-time models of latent stochastic dynamical systems. *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[29] Josef Ladenbauer, Sam McKenzie, Daniel Fine English, Olivier Hagens, and Srdjan Ostojic. Inferring and validating mechanistic models of neural microcircuits based on spike-train data. *Nature Communications*, 10(4933), 2019.

[30] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical neural population models through nonlinear embeddings. In *Advances in neural information processing systems*, pages 163–171, 2016.

[31] Yuan Zhao and Il Memming Park. Recursive variational bayesian dual estimation for nonlinear dynamics and non-gaussian observations. *stat*, 1050:27, 2017.

[32] Gabriel Barello, Adam Charles, and Jonathan Pillow. Sparse-coding variational autoencoders. *bioRxiv*, page 399246, 2018.

[33] Chethan Pandarinath, Daniel J O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky, Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg, et al. Inferring single-trial neural population dynamics using sequential autoencoders. *Nature methods*, page 1, 2018.

[34] Alexander B Wiltschko, Matthew J Johnson, Giuliano Iurilli, Ralph E Peterson, Jesse M Katon, Stan L Pashkovski, Victoria E Abraira, Ryan P Adams, and Sandeep Robert Datta. Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, 2015.

[35] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.

[36] Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall, Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John Cunningham, et al. Behavenet: nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural Information Processing Systems*, 2019.

[37] Liam Paninski and John P Cunningham. Neural data science: accelerating the experiment-analysis-theory cycle in large-scale neuroscience. *Current opinion in neurobiology*, 50:232–241, 2018.

[38] Cristopher M Niell and Michael P Stryker. Modulation of visual responses by behavioral state in mouse visual cortex. *Neuron*, 65(4):472–479, 2010.

[39] Aman B Saleem, Aslı Ayaz, Kathryn J Jeffery, Kenneth D Harris, and Matteo Carandini. Integration of visual motion and locomotion in mouse visual cortex. *Nature neuroscience*, 16(12):1864–1869, 2013.

[40] Simon Musall, Matthew T Kaufman, Ashley L Juavinett, Steven Gluf, and Anne K Churchland. Single-trial neural dynamics are dominated by richly varied movements. *Nature neuroscience*, 22(10):1677–1686, 2019.

[41] Peter Dayan, Laurence F Abbott, et al. Theoretical neuroscience: computational and mathematical modeling of neural systems. *Journal of Cognitive Neuroscience*, 15(1):154–155, 2003.

[42] Eugene M Izhikevich. *Dynamical systems in neuroscience*. MIT press, 2007.

[43] Scott A Sisson, Yanan Fan, and Mark M Tanaka. Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 104(6):1760–1765, 2007.

[44] Juliane Liepe, Paul Kirk, Sarah Filippi, Tina Toni, Chris P Barnes, and Michael PH Stumpf. A framework for parameter estimation and model selection from experimental data in systems biology using approximate bayesian computation. *Nature protocols*, 9(2):439–456, 2014.

[45] Pedro J Gonçalves, Jan-Matthis Lueckmann, Michael Deistler, Marcel Nonnenmacher, Kaan Öcal, Giacomo Bassetto, Chaitanya Chintaluri, William F Podlaski, Sara A Haddad, Tim P Vogels, et al. Training deep neural density estimators to identify mechanistic models of neural dynamics. *bioRxiv*, page 838383, 2019.

[46] George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 837–848. PMLR, 2019.

[47] Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free mcmc with amortized approximate ratio estimators. In *International Conference on Machine Learning*, pages 4239–4248. PMLR, 2020.

[48] Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow networks. *International Conference on Learning Representations*, 2017.

[49] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *International Conference on Machine Learning*, 2015.

[50] Mark S Goldman, Jorge Golowasch, Eve Marder, and LF Abbott. Global structure, robustness, and modulation of neuronal models. *Journal of Neuroscience*, 21(14):5229–5238, 2001.

[51] Gabrielle J Gutierrez, Timothy O'Leary, and Eve Marder. Multiple mechanisms switch an electrically coupled, synaptically inhibited neuron between competing rhythmic oscillators. *Neuron*, 77(5):845–858, 2013.

[52] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of selective amplification of neural activity patterns. *Neuron*, 61(4):635–648, 2009.

[53] Guillaume Hennequin, Tim P Vogels, and Wulfram Gerstner. Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron*, 82(6):1394–1406, 2014.

[54] Giulio Bondanelli, Thomas Deneux, Brice Bathellier, and Srdjan Ostojic. Population coding and network dynamics during off responses in auditory cortex. *BioRxiv*, page 810655, 2019.

[55] Ashok Litwin-Kumar, Robert Rosenbaum, and Brent Doiron. Inhibitory stabilization and visual coding in cortical circuits with multiple interneuron subtypes. *Journal of neurophysiology*, 115(3):1399–1409, 2016.

[56] Agostina Palmigiano, Francesco Fumarola, Daniel P Mossing, Nataliya Kraynyukova, Hillel Adesnik, and Kenneth Miller. Structure and variability of optogenetic responses identify the operating regime of cortex. *bioRxiv*, 2020.

[57] Chunyu A Duan, Marino Pagan, Alex T Piet, Charles D Kopec, Athena Akrami, Alexander J Riordan, Jeffrey C Erlich, and Carlos D Brody. Collicular circuits for flexible sensorimotor routing. *bioRxiv*, page 245613, 2018.

[58] Eve Marder and Vatsala Thirumalai. Cellular, synaptic and network effects of neuromodulation. *Neural Networks*, 15(4-6):479–493, 2002.

[59] Catherine Morris and Harold Lecar. Voltage oscillations in the barnacle giant muscle fiber. *Biophysical journal*, 35(1):193–213, 1981.

[60] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *Proceedings of the 5th International Conference on Learning Representations*, 2017.

[61] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017.

[62] Mark S Goldman. Memory without feedback in a neural network. *Neuron*, 61(4):621–634, 2009.

[63] Giulio Bondanelli and Srdjan Ostojic. Coding with transient trajectories in recurrent neural networks. *PLoS computational biology*, 16(2):e1007655, 2020.

[64] David Sussillo. Neural circuits as computational dynamical systems. *Current opinion in neurobiology*, 25:156–163, 2014.

[65] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current opinion in neurobiology*, 46:1–6, 2017.

[66] Abigail A Russo, Sean R Bittner, Sean M Perkins, Jeffrey S Seely, Brian M London, Antonio H Lara, Andrew Miri, Najja J Marshall, Adam Kohn, Thomas M Jessell, et al. Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, 97(4):953–966, 2018.

[67] Scott A Sisson, Yanan Fan, and Mark Beaumont. *Handbook of approximate Bayesian computation*. CRC Press, 2018.

[68] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 2020.

[69] Eve Marder and Allen I Selverston. *Dynamic biological networks: the stomatogastric nervous system*. MIT press, 1992.

[70] Hirofumi Ozeki, Ian M Finn, Evan S Schaffer, Kenneth D Miller, and David Ferster. Inhibitory stabilization of the cortical network underlies visual surround suppression. *Neuron*, 62(4):578–592, 2009.

[71] Daniel B Rubin, Stephen D Van Hooser, and Kenneth D Miller. The stabilized supralinear network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron*, 85(2):402–417, 2015.

[72] Henry Markram, Maria Toledo-Rodriguez, Yun Wang, Anirudh Gupta, Gilad Silberberg, and Caizhi Wu. Interneurons of the neocortical inhibitory system. *Nature reviews neuroscience*, 5(10):793, 2004.

[73] Bernardo Rudy, Gordon Fishell, SooHyun Lee, and Jens Hjerling-Leffler. Three groups of interneurons account for nearly 100% of neocortical gabaergic neurons. *Developmental neurobiology*, 71(1):45–61, 2011.

[74] Robin Tremblay, Soohyun Lee, and Bernardo Rudy. GABAergic Interneurons in the Neocortex: From Cellular Properties to Circuits. *Neuron*, 91(2):260–292, 2016.

[75] Carsten K Pfeffer, Mingshan Xue, Miao He, Z Josh Huang, and Massimo Scanziani. Inhibition of inhibition in visual cortex: the logic of connections between molecularly distinct interneurons. *Nature Neuroscience*, 16(8):1068, 2013.

[76] João D Semedo, Amin Zandvakili, Christian K Machens, M Yu Byron, and Adam Kohn. Cortical areas interact through a communication subspace. *Neuron*, 102(1):249–259, 2019.

[77] Daniel J Felleman and David C Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47, 1991.

[78] Guillaume Hennequin, Yashar Ahmadian, Daniel B Rubin, Mátè Lengyel, and Kenneth D Miller. The dynamical regime of sensory cortex: stable dynamics around a single stimulus-tuned attractor account for patterns of noise variability. *Neuron*, 98(4):846–860, 2018.

[79] Chunyu A Duan, Jeffrey C Erlich, and Carlos D Brody. Requirement of prefrontal and midbrain regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503, 2015.

[80] Mark A Beaumont, Wenyang Zhang, and David J Balding. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

[81] Paul Marjoram, John Molitor, Vincent Plagnol, and Simon Tavaré. Markov chain monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328, 2003.

[82] Lawrence Saul and Michael Jordan. A mean field learning algorithm for unsupervised neural networks. In *Learning in graphical models*, pages 541–554. Springer, 1998.

[83] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[84] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

[85] Mark Girolami and Ben Calderhead. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2):123–214, 2011.

[86] Andrew Golightly and Darren J Wilkinson. Bayesian parameter inference for stochastic biochemical network models using particle markov chain monte carlo. *Interface focus*, 1(6):807–820, 2011.

[87] Dustin Tran, Rajesh Ranganath, and David Blei. Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems*, pages 5523–5533, 2017.

[88] Sean R Bittner, Agostina Palmigiano, Kenneth D Miller, and John P Cunningham. Degenerate solution networks for theoretical neuroscience. *Computational and Systems Neuroscience Meeting (COSYNE), Lisbon, Portugal*, 2019.

[89] Sean R Bittner, Alex T Piet, Chunyu A Duan, Agostina Palmigiano, Kenneth D Miller, Carlos D Brody, and John P Cunningham. Examining models in theoretical neuroscience with degenerate solution networks. *Bernstein Conference 2019, Berlin, Germany*, 2019.

[90] Marcel Nonnenmacher, Pedro J Goncalves, Giacomo Bassetto, Jan-Matthis Lueckmann, and Jakob H Macke. Robust statistical inference for simulation-based models in neuroscience. In *Bernstein Conference 2018, Berlin, Germany*, 2018.

[91] Deistler Michael, , Pedro J Goncalves, Kaan Oecal, and Jakob H Macke. Statistical inference for analyzing sloppiness in neuroscience models. In *Bernstein Conference 2019, Berlin, Germany*, 2019.

[92] Jan-Matthis Lueckmann, Pedro J Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Advances in Neural Information Processing Systems*, pages 1289–1299, 2017.

[93] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

[94] Sean R Bittner and John P Cunningham. Approximating exponential family models (not single distributions) with a two-network architecture. *arXiv preprint arXiv:1903.07515*, 2019.

[95] Johan Karlsson, Milena Anguelova, and Mats Jirstrand. An efficient method for structural identifiability analysis of large dynamic systems. *IFAC Proceedings Volumes*, 45(16):941–946, 2012.

[96] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in neural information processing systems*, pages 6571–6583, 2018.

[97] Xuechen Li, Ting-Kam Leonard Wong, Ricky TQ Chen, and David Duvenaud. Scalable gradients for stochastic differential equations. *arXiv preprint arXiv:2001.01328*, 2020.

[98] Andreas Raue, Clemens Kreutz, Thomas Maiwald, Julie Bachmann, Marcel Schilling, Ursula Klingmüller, and Jens Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, 2009.

[99] Dhruva V Raman, James Anderson, and Antonis Papachristodoulou. Delineating parameter unidentifiabilities in complex models. *Physical Review E*, 95(3):032314, 2017.

[100] Maria Pia Saccomani, Stefania Audoly, and Leontina D'Angiò. Parameter identifiability of nonlinear systems: the role of initial conditions. *Automatica*, 39(4):619–632, 2003.

[101] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *arXiv preprint arXiv:1912.02762*, 2019.

[102] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in neural information processing systems*, pages 10215–10224, 2018.

[103] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29:4743–4751, 2016.

[104] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015.

[105] Emmanuel Klinger, Dennis Rickert, and Jan Hasenauer. pyabc: distributed, likelihood-free inference. *Bioinformatics*, 34(20):3591–3593, 2018.

[106] David S Greenberg, Marcel Nonnenmacher, and Jakob H Macke. Automatic posterior transformation for likelihood-free inference. *International Conference on Machine Learning*, 2019.

5 Methods

5.1 Emergent property inference (EPI)

Determining the combinations of model parameters that can produce observed data or a desired output is a key part of scientific practice. Solving inverse problems is especially important in neuroscience, since we require complex models to describe the complex phenomena of neural computations. While much machine learning research has focused on how to find latent structure in large-scale neural datasets, less has focused on inverting theoretical circuit models conditioned upon the emergent phenomena they produce. Here, we introduce a novel method for statistical inference, which finds distributions of parameter solutions that only produce the desired emergent property. This method seamlessly handles neural circuit models with stochastic nonlinear dynamical generative processes, which are predominant in theoretical neuroscience.

Consider model parameterization $\mathbf{z}$, which is a collection of scientifically interesting variables that govern the complex simulation of data $\mathbf{x}$. For example (see Section 3.1), $\mathbf{z}$ may be the electrical conductance parameters of an STG subcircuit, and $\mathbf{x}$ the evolving membrane potentials of the five neurons. In terms of statistical modeling, this circuit model has an intractable likelihood $p(\mathbf{x} \mid \mathbf{z})$, which is predicated by the stochastic differential equations that define the model. Even so, we do not scientifically reason about how $\mathbf{z}$ governs all of $\mathbf{x}$, but rather specific phenomena that are a function of the data $f(\mathbf{x}; \mathbf{z})$. In the STG example, $f(\mathbf{x}; \mathbf{z})$ measures hub neuron frequency from the evolution of $\mathbf{x}$ governed by $\mathbf{z}$. With EPI, we learn distributions of $\mathbf{z}$ that results in an average and variance of $f(\mathbf{x}; \mathbf{z})$, denoted $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$. We refer to the collection of these statistical moments as an emergent property. Such emergent properties $\mathcal{X}$ are defined through choice of $f(\mathbf{x}; \mathbf{z})$ (which may be one or multiple statistics), $\boldsymbol{\mu}$, and $\boldsymbol{\sigma}^2$

$$\mathcal{X} : \mathbb{E}_{\mathbf{z},\mathbf{x}} \left[f(\mathbf{x}; \mathbf{z}) \right] = \boldsymbol{\mu}, \mathrm{Var}_{\mathbf{z},\mathbf{x}} \left[f(\mathbf{x}; \mathbf{z}) \right] = \boldsymbol{\sigma}^2. \tag{11}$$

Precisely, the emergent property statistics $f(\mathbf{x}; \mathbf{z})$ must have means $\boldsymbol{\mu}$ and variances $\boldsymbol{\sigma}^2$ over the EPI distribution of parameters and stochasticity of the data given the parameters. By defining these means and variances over both levels of stochasticity – the inferred distribution and that of the model – there is a fine degree of control over predictions made by the inferred parameters.

In EPI, deep probability distributions are optimized to learn the inferred distribution. In deep probability distributions, a simple random variable $\mathbf{z}_0 \sim q_0(\mathbf{z}_0)$ is mapped deterministically via a sequence of deep neural network layers (g_1, .. g_l) parameterized by weights and biases $\boldsymbol{\theta}$ to the support of the distribution of interest:

$$\mathbf{z} = g_{\boldsymbol{\theta}}(\mathbf{z}_0) = g_l(..g_1(\mathbf{z}_0)) \sim q_{\boldsymbol{\theta}}(\mathbf{z}). \tag{12}$$

Such deep probability distributions embed the inferred distribution in a deep network. Once optimized, this deep network representation has remarkably useful properties: fast sampling, probability evaluations, and also first- and second-order probability gradient evaluations.

By choosing a neural circuit model, often represented as a system of differential equations, we implictly definge a model likelihood $p(\mathbf{x} \mid \mathbf{z})$, which may be unknown or intractable for our purposes. Given this model choice and that of an emergent property $\mathcal{X}$, $q_{\boldsymbol{\theta}}(\mathbf{z})$ is optimized via the neural network parameters $\boldsymbol{\theta}$ to find a maximally entropic distribution $q^*_{\boldsymbol{\theta}}$ within the deep variational family $\mathcal{Q}$ producing the emergent property $\mathcal{X}$:

$$\begin{aligned} q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}) = q^*_{\boldsymbol{\theta}}(\mathbf{z}) = \underset{q_{\boldsymbol{\theta}} \in Q}{\operatorname{argmax}} \, H(q_{\boldsymbol{\theta}}(\mathbf{z})) \\ \text{s.t. } \mathcal{X} \quad : \quad \mathbb{E}_{\mathbf{z},\mathbf{x}}\left[f(\mathbf{x};\mathbf{z})\right] = \boldsymbol{\mu}, \mathrm{Var}_{\mathbf{z},\mathbf{x}}\left[f(\mathbf{x};\mathbf{z})\right] = \boldsymbol{\sigma}^2. \end{aligned} \tag{13}$$

Entropy is chosen as the normative selection principle to match that of Bayesian inference (see Section 5.1.5). However, a key difference is that variational inference and other Bayesian methods do not constrain the predictions of their inferred parameter distribution. This optimization is executed using the algorithm of Maximum Entropy Flow Networks (MEFNs) [48].

In the remainder of Section 5.1, we will explain the finer details and motivation of the EPI method. First, we explain related approaches and what EPI introduces to this domain (Section 5.1.1). Second, we describe the special class of deep probability distributions used in EPI called normalizing flows (Section 5.1.2). Next, we explain the constrained optimization technique used to solve Equation 13 (Section 5.1.3). Then, we demonstrate the details of this optimization in a toy example (Section 5.1.4). Finally, we establish the known relationship between maximum entropy distributions and exponential families (Section 5.1.5), which is used to explain how EPI can be viewed as a form of variational inference (Section 5.1.6).

5.1.1 Related approaches

When Bayesian inference problems lack conjugacy, scientists use approximate inference methods like variational inference (VI) [82] and Markov chain Monte Carlo (MCMC) [83,84]. After optimization, variational methods return a parameterized posterior distribution, which we can analyze. Also, the variational approximating distribution class is often chosen such that it permits fast sampling. In contrast MCMC methods only produce samples from the approximated posterior distribution. No parameterized distribution is estimated, and additional samples are always generated with the same sampling complexity. Inference in models defined by systems of differential has been demonstrated with MCMC [85], although this approach requires tractable likelihoods. Advancements have leveraged structure in stochastic differential equation models to improve likelihood approximations, thus expanding the domain of applicable models [86].

Simulation-based inference [68] is model parameter inference in the absence of a tractable likelihood function. The most prevalent approach to simulation-based inference is approximate Bayesian computation [80], in which satisfactory parameter samples are kept from random prior sampling according to a rejection heuristic. The obtained set of parameters do not have a probabilities, and further insight about the model must be gained from examination of the parameter set and their generated activity. Methodological advances to ABC methods have come through the use of Markov chain Monte Carlo (MCMC-ABC) [81] and sequential Monte Carlo (SMC-ABC) [43] sampling techniques. SMC-ABC is considered state-of-the-art ABC, yet this approach still struggles to scale in dimensionality (cf. Fig. 2). Furthermore, once a parameter set has been obtained by SMC-ABC from a finite set of particles, the SMC-ABC algorithm must be run again from scratch with a new population of initialized particles to obtain additional samples.

For scientific model analysis, we seek a parameter distribution exhibiting the properties of a well-chosen variational approximation: a parametric form conferring analytic calculations, and trivial sampling time. For this reason, ABC and MCMC techniques are unattractive, since they only produce a set of parameter samples and have unchanging sampling rate. EPI infers parameters in mechanistic models using the MEFN [48] algorithm using a deep variational approximation. The deep neural network of EPI defines the parametric form of the distribution approximation. Furthermore, the EPI distribution is constrained to produce an emergent property. In other words, the summary statistics of the posterior predictive distribution are fixed to have certain first and second moments. EPI optimization is enabled using stochastic gradient techniques in the spirit of likelihood-free variational inference [87]. The analytic relationship between EPI and variational inference is explained in Secton 5.1.6.

We note that, during our preparation and early presentation of this work [88, 89], another work has arisen with broadly similar goals: bringing statistical inference to mechanistic models of neural circuits ([45, 90, 91]).We are encouraged by this general problem being recognized by others in the community, and we emphasize that these works offer complementary neuroscientific contributions (different theoretical models of focus) and use different technical methodologies (ours is built on our prior work [48], theirs similarly [92]).

The method EPI differs from SNPE in some key ways. SNPE belongs to a "sequential" class of recently developed simulation-based inference methods in which two neural networks are used for posterior inference. This first neural network is a deep probability distribution (normalizing flow) used to estimate the posterior $p(\mathbf{z} \mid \mathbf{x})$ (SNPE) or the likelihood $p(\mathbf{x} \mid \mathbf{z})$ (sequential neural likelihood (SNL [46])). A recent advance uses an unconstrained neural network to estimate the likelihood ratio (sequential neural ratio estimation (SNRE [47]). In SNL and SNRE, MCMC sampling techniques are used to obtain samples from the approximated posterior. This contrasts with EPI and SNPE, which use deep probability distributions to model parameters, which facilitates immediate measurements of sample probability, gradient, or Hessian for system analysis. The second neural network in this sequential class of methods is the amortizer. This unconstrained deep network maps data $\mathbf{x}$ (or statistics $f(\mathbf{x}; \mathbf{z})$ or model parameters $\mathbf{z}$ to the weights and biases of the first neural network. These methods are optimized on a conditional density (or ratio) estimation objective. The data used to optimize this objective are generated via an adaptive procedure, in which training data pairs ($\mathbf{x}_i$, $\mathbf{z}_i$) become sequentiall closer to the true data and posterior.

The approximating fidelity of the deep probability distribution in sequential approaches is optimized to generalize across the training distribution of the conditioning variable. This generalization property of the sequential methods can reduce the accuracy at the singular posterior of interest. Whereas in EPI, the entire expressivity of the deep probability distribution is dedicated to learning a single distribution as well as possible. Amortization is not possible in EPI, since EPI learns an exponential family distribution parameterized by its mean (see Section 5.1.5). Since EPI distributions are defined by the mean $\boldsymbol{\mu}$ of their statistics, there is the well-known inverse mapping problem of exponential families [93] that prohibits an amortization based approach. However, we have shown that the same two-network architecture of the sequential simulation-based inference methods can be used for amortized inference in intractable exponential family posteriors using their natural parameterization [94].

Finally, one important differentiating factor between EPI and sequential simulation-based inference methods is that EPI leverages gradients $\nabla_{\mathbf{z}} f(\mathbf{x}; \mathbf{z})$ during optimization. These gradients can improve convergence time and scalability, as we have shown on an example conditioning low-rank RNN connectivity on the property of stable amplification (see Section 3.3). With EPI, we prove out the suggestion that a deep inference technique can improve efficiency by leveraging these model gradients when they are tractable. Sequential simulation-based inference techniques may be better suited for sceintific problems where $\nabla_{\mathbf{z}} f(\mathbf{x};\mathbf{z})$ is intractable or unavailable: when there is a non-differentiable model or it requires lengthy simulations. However, the sequential simulation-based inference techniques cannot constrain the predictions of the inferred distribution in the manner of EPI.

Structural identifiability analysis involves the measurement of sensitivity and unidentifiabilities in natural models. Around a point, one can measure the Jacobian. One approach that scales well is EAR [95]. A popular efficient approach for systems of ODEs has been neural ODE adjoint [96] and its stochastic adaptation [97]. Casting identifiability as a statistical estimation problem, the profile likelihood can assess via iterated optimization while holding parameters fixed [98]. An exciting recent method is capable of recovering the functional form of such unidentifiabilities away from a point by following degenerate dimensions of the fisher information matrix [99]. Global structural non-identifiabilities can be found for models with polynomial or rational dynamics equations using DAISY [100]. With EPI, we have all the benefits given by a statistical inference method plus the ability to query the first- or second-order gradient of the probability of the inferred distribution at any chosen parameter value. The second-order gradient of the log probability (the Hessian), which is directly afforded by EPI distributions, produces salient information about parametric sensitivty of the emergent property. For example, the eigenvector with most negative eigenvalue of the Hessian shows parametric combinations away from a parameter choice that decrease the in EPI distribution probability the fastest. We refer to this eigenvector as the sensitivity dimension, and it is used to generate scientific insight about a model of superior colliculus connectivity (see Section 3.5).

5.1.2 Deep probability distributions and normalizing flows

Deep probability distributions are comprised of multiple layers of fully connected neural networks (Equation 12). When each neural network layer is restricted to be a bijective function, the sample density can be calculated using the change of variables formula at each layer of the network. For $\mathbf{z}_i = g_i(\mathbf{z_{i-1}})$,

$$p(\mathbf{z}_i) = p(g_i^{-1}(\mathbf{z}_i)) \left| \det \frac{\partial g_i^{-1}(\mathbf{z}_i)}{\partial \mathbf{z}_i} \right| = p(\mathbf{z}_{i-1}) \left| \det \frac{\partial g_i(\mathbf{z}_{i-1})}{\partial \mathbf{z}_{i-1}} \right|^{-1}. \tag{14}$$

However, this computation has cubic complexity in dimensionality for fully connected layers. By restricting our layers to normalizing flows [49, 101] – bijective functions with fast log determinant Jacobian computations, which confer a fast calculation of the sample log probability. Fast log probability calculation confers efficient optimization of the maximum entropy objective (see Section 5.1.3). We use the Real NVP [60] normalizing flow class, because its coupling architecture confers both fast sampling (forward) and fast log probability evaluation (backward). Fast probability evaluation in turn facilitates fast gradient and Hessian evaluation of log probability throughout parameter space. Glow permutations were used in between coupling stages [102]. This is in contrast to autoregressive architectures [61, 103], in which only one of the forward or backward passes can be efficient. In this work, normalizing flows are used as flexible posterior approximations $q_{\boldsymbol{\theta}}(\mathbf{z})$ having weights and biases $\boldsymbol{\theta}$. We specify the architecture used in each application by the number of Real-NVP affine coupling stages, and the number of neural network layers and units per layer of the conditioning functions.

5.1.3 Augmented Lagrangian optimization

To optimize $q_{\boldsymbol{\theta}}(\mathbf{z})$ in Equation 13, the constrained maximum entropy optimization is executed using the augmented Lagrangian method. The following objective is minimized:

$$L(\boldsymbol{\theta}; \boldsymbol{\eta}_{\text{opt}}, c) = -H(q_{\boldsymbol{\theta}}) + \boldsymbol{\eta}_{\text{opt}}^\top R(\boldsymbol{\theta}) + \frac{c}{2}||R(\boldsymbol{\theta})||^2 \tag{15}$$

where average constraint violations $R(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{z}\sim q_{\boldsymbol{\theta}}}\left[\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}|\mathbf{z})}\left[T(\mathbf{x};\mathbf{z}) - \boldsymbol{\mu}_{\text{opt}}\right]\right]$, $\boldsymbol{\eta}_{\text{opt}} \in \mathbb{R}^m$ are the Lagrange multipliers where $m = |\boldsymbol{\mu}_{\text{opt}}| = |T(\mathbf{x};\mathbf{z})| = 2|f(\mathbf{x};\mathbf{z})|$, and c is the penalty coefficient. The sufficient statistics $T(\mathbf{x};\mathbf{z})$ and mean parameter $\boldsymbol{\mu}_{\text{opt}}$ are determined by the means $\boldsymbol{\mu}$ and variances $\boldsymbol{\sigma}^2$ of emergent property statistics $f(\mathbf{x};\mathbf{z})$ defined in Equation 13 (see Section 5.1.6). Specifically, $T(\mathbf{x};\mathbf{z})$ is a concatenation of the first and second moments, $\boldsymbol{\mu}_{\text{opt}}$ is a concatenation of $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ (see section 5.1.5), and the Lagrange multipliers are closely related to the natural parameters $\boldsymbol{\eta}$ of exponential families (see Section 5.1.5). Weights and biases $\boldsymbol{\theta}$ of the deep probability distribution are optimized according to Equation 15 using the Adam optimizer with learning rate 10^{-3} [104].

The gradient with respect to entropy $H(q_{\boldsymbol{\theta}}(\mathbf{z}))$ can be expressed using the reparameterization trick as an expectation of the negative log density of parameter samples $\mathbf{z}$ over the randomness in the parameterless initial distribution $q_0(\mathbf{z}_0)$:

$$H(q_{\boldsymbol{\theta}}(\mathbf{z})) = \int -q_{\boldsymbol{\theta}}(\mathbf{z})\log(q_{\boldsymbol{\theta}}(\mathbf{z}))d\mathbf{z} = \mathbb{E}_{\mathbf{z}\sim q_{\boldsymbol{\theta}}}\left[-\log(q_{\boldsymbol{\theta}}(\mathbf{z}))\right] = \mathbb{E}_{\mathbf{z}_0\sim q_0}\left[-\log(q_{\boldsymbol{\theta}}(g_{\boldsymbol{\theta}}(\mathbf{z}_0)))\right]. \tag{16}$$

Thus, the gradient of the entropy of the deep probability distribution can be estimated as an average with respect to the base distribution $\mathbf{z}_0$:

$$\nabla_{\boldsymbol{\theta}} H(q_{\boldsymbol{\theta}}(\mathbf{z})) = \mathbb{E}_{\mathbf{z}_0\sim q_0}\left[-\nabla_{\boldsymbol{\theta}}\log(q_{\boldsymbol{\theta}}(g_{\boldsymbol{\theta}}(\mathbf{z}_0)))\right]. \tag{17}$$

The lagrangian parameters $\boldsymbol{\eta}_{\text{opt}}$ are initialized to zero and adapted following each augmented Lagrangian epoch, which is a period of optimization with fixed ($\boldsymbol{\eta}_{\text{opt}}$, c) for a given number of stochastic optimization iterations. A low value of c is used initially, and conditionally increased after each epoch based on constraint error reduction. The penalty coefficient is updated based on the result of a hypothesis test regarding the reduction in constraint violation. The p-value of $\mathbb{E}[||R(\boldsymbol{\theta}_{k+1})||] > \gamma \mathbb{E}\left[||R(\boldsymbol{\theta}_k)||\right]$ is computed, and c_{k+1} is updated to βc_k with probability $1-p$. The other update rule is $\boldsymbol{\eta}_{\text{opt},k+1} = \boldsymbol{\eta}_{\text{opt},k} + c_k \frac{1}{n} \sum_{i=1}^{n} (T(\mathbf{x}^{(i)}) - \boldsymbol{\mu}_{\text{opt}})$ given a batch size n. Throughout the study, $\gamma = 0.25$, while β was chosen to be either 2 or 4. The batch size of EPI also varied according to application.

The intention is that c and $\boldsymbol{\eta}_{\text{opt}}$ start at values encouraging entropic growth early in optimization. With each training epoch in which the update rule for c is invoked by unsatisfactory constraint error reduction, the constraint satisfaction terms are increasingly weighted, resulting in a decreased entropy. This encourages the discovery of suitable regions of parameter space, and the subsequent refinement of the distribution to produce the emergent property (see example in Section 5.1.4). The momentum parameters of the Adam optimizer are reset at the end of each augmented Lagrangian epoch.

Rather than starting optimization from some $\boldsymbol{\theta}$ drawn from a randomized distribution, we found that initializing $q_{\boldsymbol{\theta}}(\mathbf{z})$ to approximate an isotropic Gaussian distribution conferred more stable, consistent optimization. The parameters of the Gaussian initialization were chosen on an application-specific basis. Throughout the study, we chose isotropic Gaussian initializations with mean $\boldsymbol{\mu}_{\text{init}}$ at the center of the distribution support and some standard deviation $\boldsymbol{\sigma}_{\text{init}}$, except for one case, where an initialization informed by random search was used (see Section 5.2.1).

To assess whether the EPI distribution $q_{\boldsymbol{\theta}}(\mathbf{z})$ produces the emergent property, we assess whether each individual constraint on the means and variances of $f(\mathbf{x}; \mathbf{z})$ is satisfied. We consider the EPI to have converged when a null hypothesis test of constraint violations $R(\boldsymbol{\theta})_i$ being zero is accepted for all constraints $i \in \{1, ..., m\}$ at a significance threshold $\alpha = 0.05$. This significance threshold is adjusted through Bonferroni correction according to the number of constraints m. The p-values for each constraint are calculated according to a two-tailed nonparametric test, where 200 estimations of the sample mean $R(\boldsymbol{\theta})^i$ are made using N_{test} samples of $\mathbf{z} \sim q_{\boldsymbol{\theta}}(\mathbf{z})$ at the end of the augmented Lagrangian epoch.

When assessing the suitability of EPI for a particular modeling question, there are some important technical considerations. First and foremost, as in any optimization problem, the defined emergent property should always be appropriately conditioned (constraints should not have wildly different units). Furthermore, if the program is underconstrained (not enough constraints), the distribution grows (in entropy) unstably unless mapped to a finite support. If overconstrained, there is no parameter set producing the emergent property, and EPI optimization will fail (appropriately). Next, one should consider the computational cost of the gradient calculations. In the best circumstance, there is a simple, closed form expression (e.g. Section 5.2.2) for the emergent property statistic given the model parameters. On the other end of the spectrum, many forward simulation iterations may be required before a high quality measurement of the emergent property statistic is available (e.g. Section 5.2.1). In such cases, backpropagating gradients through the SDE evolution will be expensive.

5.1.4 Example: 2D LDS

To gain intuition for EPI, consider a two-dimensional linear dynamical system (2D LDS) model (Fig. S1A):

$$\tau \frac{d\mathbf{x}}{dt} = A\mathbf{x} \tag{18}$$

with

$$A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix}. \tag{19}$$

To run EPI with the dynamics matrix elements as the free parameters $\mathbf{z} = [a_1, a_2, a_3, a_4]$ (fixing $\tau = 1$), the emergent property statistics $T(\mathbf{x})$ were chosen to contain the first and second moments of the oscillatory frequency, $\frac{\text{imag}(\lambda_1)}{2\pi}$, and the growth/decay factor, real(λ_1), of the oscillating system. λ_1 is the eigenvalue of greatest real part when the imaginary component is zero, and alternatively of positive imaginary component when the eigenvalues are complex conjugate pairs. To learn the distribution of real entries of A that produce a band of oscillating systems around 1Hz, we formalized this emergent property as real(λ_1) having mean zero with variance 0.25^2, and the oscillation frequency $2\pi\text{imag}(\lambda_1)$ having mean $\omega = 1$ Hz with variance $(0.1\text{Hz})^2$:

$$\mathbb{E}\left[T(\mathbf{x})\right] \triangleq \mathbb{E}\begin{bmatrix} \text{real}(\lambda_1) \\ \text{imag}(\lambda_1) \\ (\text{real}(\lambda_1) - 0)^2 \\ (\text{imag}(\lambda_1) - 2\pi\omega)^2 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 2\pi\omega \\ 0.25^2 \\ (2\pi 0.1)^2 \end{bmatrix} \triangleq \boldsymbol{\mu}. \tag{20}$$

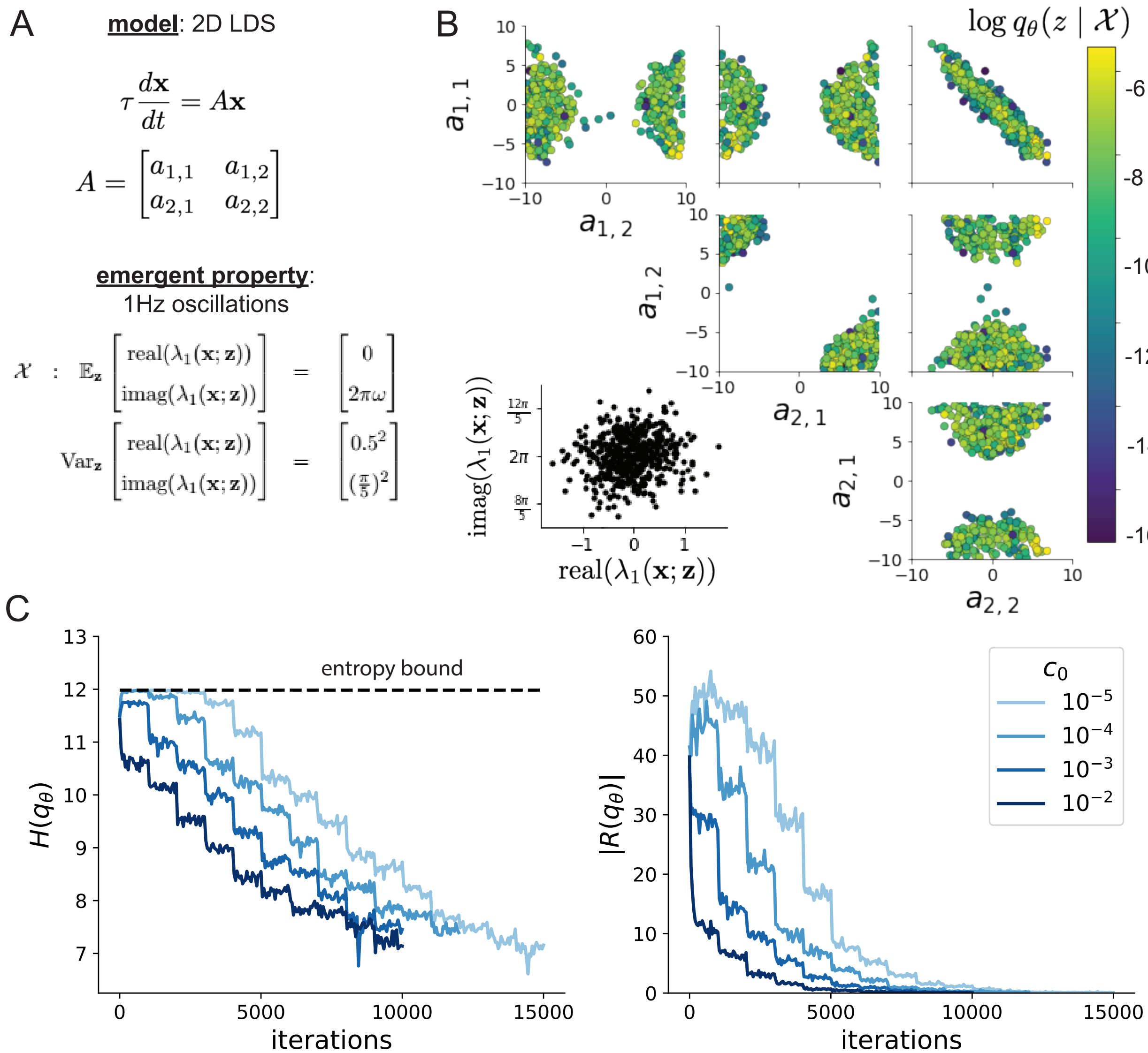

Figure 6: (LDS1): **A**. Two-dimensional linear dynamical system model, where real entries of the dynamics matrix A are the parameters. **B**. The EPI distribution for a two-dimensional linear dynamical system with $\tau = 1$ that produces an average of 1Hz oscillations with some small amount of variance. Dashed lines indicate the parameter axes. **C**. Entropy throughout the optimization. At the beginning of each augmented Lagrangian epoch (2,000 iterations), the entropy dipped due to the shifted optimization manifold where emergent property constraint satisfaction is increasingly weighted. **D**. Emergent property moments throughout optimization. At the beginning of each augmented Lagrangian epoch, the emergent property moments adjust closer to their constraints.

Unlike the models we presented in the main text, this model admits an analytical form for the mean emergent property statistics given parameter $\mathbf{z}$, since the eigenvalues can be calculated using the quadratic formula:

$$\lambda = \frac{(\frac{a_1+a_4}{\tau}) \pm \sqrt{(\frac{a_1+a_4}{\tau})^2 + 4(\frac{a_2a_3 - a_1a_4}{\tau})}}{2}. \tag{21}$$

Importantly, even though $\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}|\mathbf{z})}\left[T(\mathbf{x})\right]$ is calculable directly via a closed form function and does not require simulation, we cannot derive the distribution $q^*_{\boldsymbol{\theta}}$ directly. This fact is due to the formally hard problem of the backward mapping: finding the natural parameters η from the mean parameters $\boldsymbol{\mu}$ of an exponential family distribution [93]. Instead, we used EPI to approximate this distribution (Fig. S1B). We used a real-NVP normalizing flow architecture with four masks, two neural network layers of 15 units per mask, with batch normalization momentum 0.99, mapped onto a support of $z_i \in [-10, 10]$. (see Section 5.1.2).

Even this relatively simple system has nontrivial (though intuitively sensible) structure in the parameter distribution. To validate our method, we analytically derived the contours of the probability density from the emergent property statistics and values. In the a_1-a_4 plane, the black line at real$(\lambda_1) = \frac{a_1+a_4}{2} = 0$, dotted black line at the standard deviation real$(\lambda_1) = \frac{a_1+a_4}{2} \pm 0.25$, and the dotted gray line at twice the standard deviation real$(\lambda_1) = \frac{a_1+a_4}{2} \pm 0.5$ follow the contour of probability density of the samples (Fig. S2A). The distribution precisely reflects the desired statistical constraints and model degeneracy in the sum of a_1 and a_4. Intuitively, the parameters equivalent with respect to emergent property statistic real(λ_1) have similar log densities.

To explain the bimodality of the EPI distribution, we examined the imaginary component of λ_1. When real$(\lambda_1) = \frac{a_1+a_4}{2} = 0$, we have

$$\text{imag}(\lambda_1) = \begin{cases} \sqrt{\frac{a_1a_4 - a_2a_3}{\tau}}, & \text{if } a_1a_4 < a_2a_3 \\ 0 & \text{otherwise} \end{cases}. \tag{22}$$

When $\tau = 1$ and $a_1a_4 > a_2a_3$ (center of distribution above), we have the following equation for the other two dimensions:

$$\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 \tag{23}$$

Since we constrained $\mathbb{E}_{\mathbf{z}\sim q_{\boldsymbol{\theta}}}\left[\text{imag}(\lambda)\right] = 2\pi$ (with $\omega = 1$), we can plot contours of the equation $\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 = (2\pi)^2$ for various a_1a_4 (Fig. S2B). With $\sigma_{1,4} = \mathbb{E}_{\mathbf{z}\sim q_{\boldsymbol{\theta}}}(|a_1a_4 - E_{q_{\boldsymbol{\theta}}}[a_1a_4]|)$, we show the contours as $a_1a_4 = 0$ (black), $a_1a_4 = -\sigma_{1,4}$ (black dotted), and $a_1a_4 = -2\sigma_{1,4}$ (grey dotted). This validates the curved structure of the inferred distribution learned through EPI. We

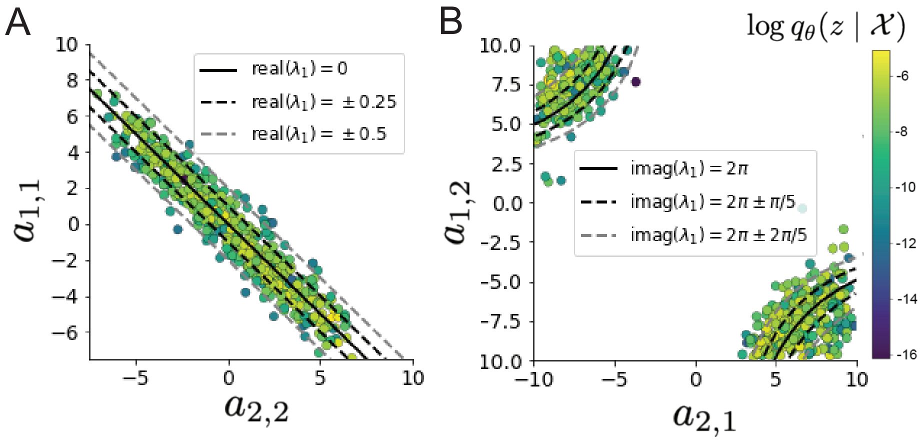

Figure 7: (LDS2): **A**. Probability contours in the a_1-a_4 plane were derived from the relationship to emergent property statistic of growth/decay factor real(λ_1). **B**. Probability contours in the a_2-a_3 plane were derived from the emergent property statistic of oscillation frequency 2πimag(λ_1).

took steps in negative standard deviation of $a_1 a_4$ (dotted and gray lines), since there are few positive values $a_1 a_4$ in the learned distribution. Subtler combinations of model and emergent property will have more complexity, further motivating the use of EPI for understanding these systems. As we expect, the distribution results in samples of two-dimensional linear systems oscillating near 1Hz (Fig. S3).

5.1.5 Maximum entropy distributions and exponential families

EPI is a maximum entropy distribution, which have fundamental links to exponential family distributions. A maximum entropy distribution of form:

$$\begin{aligned} p^*(\mathbf{z}) &= \underset{p \in \mathcal{P}}{\operatorname{argmax}}\, H(p(\mathbf{z})) \\ &\text{s.t. } \mathbb{E}_{\mathbf{z}\sim p}\left[T(\mathbf{z})\right] = \boldsymbol{\mu}_{\text{opt}}. \end{aligned} \tag{24}$$

will have probability density in the exponential family:

$$p^*(\mathbf{z}) \propto \exp(\boldsymbol{\eta}^\top T(\mathbf{z})). \tag{25}$$

The mappings between the mean parameterization $\boldsymbol{\mu}_{\text{opt}}$ and the natural parameterization $\boldsymbol{\eta}$ are formally hard to identify except in special cases [93].

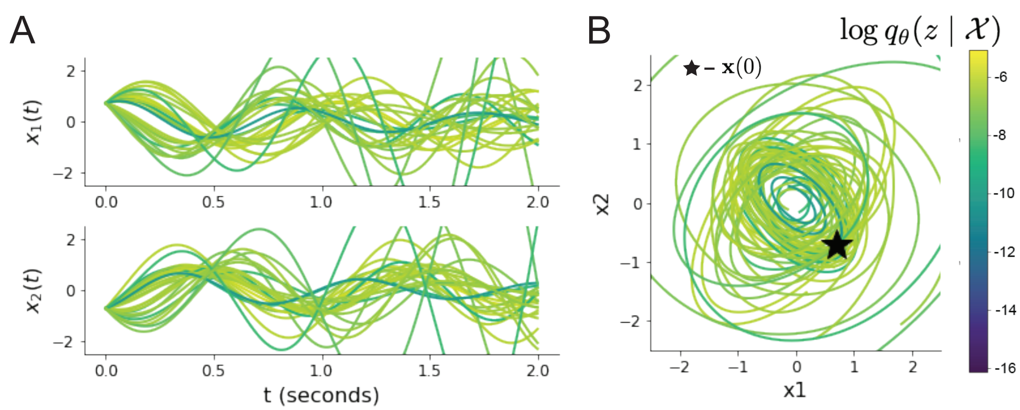

Figure 8: (LDS3): Sampled dynamical systems $\mathbf{z} \sim q_{\boldsymbol{\theta}}(\mathbf{z})$ and their simulated activity from $\mathbf{x}(0) = [\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}]$ colored by log probability. **A**. Each dimension of the simulated trajectories throughout time. **B**. The simulated trajectories in phase space.

In EPI, emergent properties are defined as statistics having a fixed mean and variance as in Equation 4. The variance constraint is a second moment constraint on $f(\mathbf{x};\mathbf{z})$

$$\text{Var}_{\mathbf{z},\mathbf{x}}\left[f(\mathbf{x};\mathbf{z})\right] = \mathbb{E}_{\mathbf{z},\mathbf{x}}\left[\left(f(\mathbf{x};\mathbf{z}) - \boldsymbol{\mu}\right)^2\right] \tag{26}$$

As a general maximum entropy distribution (Equation 24), the sufficient statistics vector contains both first and second order moments of $f(\mathbf{x};\mathbf{z})$

$$T(\mathbf{x};\mathbf{z}) = \begin{bmatrix} f(\mathbf{x};\mathbf{z}) \\ \left(f(\mathbf{x};\mathbf{z}) - \boldsymbol{\mu}\right)^2 \end{bmatrix}, \tag{27}$$

which are constrained to the chosen means and variances

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\sigma}^2 \end{bmatrix}. \tag{28}$$

5.1.6 EPI as variational inference

In Bayesian inference a prior belief about model parameters $\mathbf{z}$ is stated in a prior distribution $p(\mathbf{z})$, and the statistical model capturing the effect of $\mathbf{z}$ on observed data points $\mathbf{x}$ is formalized in the likelihood distribution $p(\mathbf{x} \mid \mathbf{z})$. In Bayesian inference, we obtain a posterior distribution $p(\mathbf{z} \mid \mathbf{x})$, which captures how the data inform our knowledge of model parameters using Bayes' rule:

$$p(\mathbf{z} \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \mathbf{z})p(\mathbf{z})}{p(\mathbf{x})}. \tag{29}$$

The posterior distribution is analytically available when the prior is conjugate with the likelihood. However, conjugacy is rare in practice, and alternative methods, such as variational inference [82], are utilized.

In variational inference, a posterior approximation $q_{\boldsymbol{\theta}}^*$ is chosen from within some variational family $\mathcal{Q}$

$$q_{\boldsymbol{\theta}}^*(\mathbf{z}) = \operatorname*{argmin}_{q_{\boldsymbol{\theta}} \in Q} KL(q_{\boldsymbol{\theta}}(\mathbf{z}) \mid\mid p(\mathbf{z} \mid \mathbf{x})). \tag{30}$$

The KL divergence can be written in terms of entropy of the variational approximation:

$$KL(q_{\boldsymbol{\theta}}(\mathbf{z}) \mid\mid p(\mathbf{z} \mid \mathbf{x})) = \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} \left[\log(q_{\boldsymbol{\theta}}(\mathbf{z}))\right] - \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} \left[\log(p(\mathbf{z} \mid \mathbf{x}))\right] \tag{31}$$

$$= -H(q_{\boldsymbol{\theta}}) - \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} \left[\log(p(\mathbf{x} \mid \mathbf{z})) + \log(p(\mathbf{z})) - \log(p(\mathbf{x}))\right] \tag{32}$$

Since the marginal distribution of the data $p(\mathbf{x})$ (or "evidence") is independent of $\boldsymbol{\theta}$, variational inference is executed by optimizing the remaining expression. This is usually framed as maximizing the evidence lower bound (ELBO)

$$\operatorname*{argmin}_{q_{\boldsymbol{\theta}} \in Q} KL(q_{\boldsymbol{\theta}} \mid\mid p(\mathbf{z} \mid \mathbf{x})) = \operatorname*{argmax}_{q_{\boldsymbol{\theta}} \in Q} H(q_{\boldsymbol{\theta}}) + \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} \left[\log(p(\mathbf{x} \mid \mathbf{z})) + \log(p(\mathbf{z}))\right]. \tag{33}$$

Now, consider the setting where we have chosen a uniform prior, and stipulate a mean-field gaussian likelihood on a chosen statistic of the data $f(\mathbf{x}; \mathbf{z})$

$$p(\mathbf{x} \mid \mathbf{z}) = \mathcal{N}(f(\mathbf{x}; \mathbf{z}) \mid \boldsymbol{\mu}_f, \Sigma_f), \tag{34}$$

where $\Sigma_f = \text{diag}(\boldsymbol{\sigma}_f^2)$. The log likelihood is then proportional to a dot product of the natural parameter of this mean-field gaussian distribution and the first and second moment statistics.

$$\log p(\mathbf{x} \mid \mathbf{z}) \propto \boldsymbol{\eta}_f^\top T(\mathbf{x}, \mathbf{z}), \tag{35}$$

where

$$\boldsymbol{\eta}_f = \begin{bmatrix} \frac{\boldsymbol{\mu}_f}{\boldsymbol{\sigma}_f^2} \\ \frac{-1}{2\boldsymbol{\sigma}_f^2} \end{bmatrix}, \text{ and} \tag{36}$$

$$T(\mathbf{x}; \mathbf{z}) = \begin{bmatrix} f(\mathbf{x}; \mathbf{z}) \\ (f(\mathbf{x}; \mathbf{z}) - \boldsymbol{\mu}_f)^2 \end{bmatrix}. \tag{37}$$

The variational objective is then

$$\operatorname*{argmax}_{q_{\boldsymbol{\theta}} \in Q} H(q_{\boldsymbol{\theta}}) + \boldsymbol{\eta}_f^\top \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} \left[T(\mathbf{x}; \mathbf{z})\right] \tag{38}$$

Comparing this to the Lagrangian objective (without augmentation) of EPI, we see they are the same

$$\begin{aligned} q_{\boldsymbol{\theta}}^*(\mathbf{z}) &= \underset{q_{\boldsymbol{\theta}} \in Q}{\operatorname{argmin}} -H(q_{\boldsymbol{\theta}}) + \boldsymbol{\eta}_{\text{opt}}^\top \left(\mathbb{E}_{\mathbf{z},\mathbf{x}} \left[T(\mathbf{x};\mathbf{z}) \right] - \boldsymbol{\mu}_{\text{opt}} \right) \\ &= \underset{q_{\boldsymbol{\theta}} \in Q}{\operatorname{argmin}} -H(q_{\boldsymbol{\theta}}) + \boldsymbol{\eta}_{\text{opt}}^\top \mathbb{E}_{\mathbf{z},\mathbf{x}} \left[T(\mathbf{x};\mathbf{z}) \right]. \end{aligned} \tag{39}$$

where $T(\mathbf{x};\mathbf{z})$ consists of the first and second moments of the emergent property statistic $f(\mathbf{x};\mathbf{z})$ (Equation 27). Thus, EPI is implicitly executing variational inference with a uniform prior and a mean-field gaussian likelihood on the emergent property statistics. The mean and variances of the mean-field gaussian likelihood are predicated by $\boldsymbol{\eta}_{\text{opt}}$ (Equations 36 and 38), which is adapted after each EPI optimization epoch based on $\mathcal{X}$ (see Section 5.1.3). In EPI, the inferred distribution is not conditioned on a finite dataset as in variational inference, but rather the emergent property $\mathcal{X}$ dictates the likelihood parameterization such that the inferred distribution will produce the emergent property. As a note, we could not simply choose $\boldsymbol{\mu}_f$ and $\boldsymbol{\sigma}_f$ directly from the outset, since we do not know which of these choices will produce the emergent property $\mathcal{X}$, which necessitates the EPI optimization routine that adapts $\boldsymbol{\eta}_{\text{opt}}$. Accordingly, we replace the notation of $p(\mathbf{z} \mid \mathbf{x})$ with $p(\mathbf{z} \mid \mathcal{X})$ conceptualizing an inferred distribution that obeys emergent property $\mathcal{X}$ (see Section 5.1).

5.2 Theoretical models

In this study, we used emergent property inference to examine several models relevant to theoretical neuroscience. Here, we provide the details of each model and the related analyses.

5.2.1 Stomatogastric ganglion

We analyze how the parameters $\mathbf{z} = [g_{\text{el}}, g_{\text{synA}}]$ govern the emergent phenomena of intermediate hub frequency in a model of the stomatogastric ganglion (STG) [51] shown in Figure 1A with activity $\mathbf{x} = [x_{\text{f1}}, x_{\text{f2}}, x_{\text{hub}}, x_{\text{s1}}, x_{\text{s2}}]$, using the same hyperparameter choices as Gutierrez et al. Each neuron's membrane potential $x_\alpha(t)$ for $\alpha \in \{\text{f1}, \text{f2}, \text{hub}, \text{s1}, \text{s2}\}$ is the solution of the following stochastic differential equation:

$$C_m \frac{dx_\alpha}{dt} = -\left[h_{leak}(\mathbf{x};\mathbf{z}) + h_{Ca}(\mathbf{x};\mathbf{z}) + h_K(\mathbf{x};\mathbf{z}) + h_{hyp}(\mathbf{x};\mathbf{z}) + h_{elec}(\mathbf{x};\mathbf{z}) + h_{syn}(\mathbf{x};\mathbf{z}) \right] + dB. \tag{40}$$

The input current of each neuron is the sum of the leak, calcium, potassium, hyperpolarization, electrical and synaptic currents as well as gaussian noise dB. Each current component is a function of all membrane potentials and the conductance parameters $\mathbf{z}$.

The capacitance of the cell membrane was set to $C_m = 1nF$. Specifically, the currents are the difference in the neuron's membrane potential and that current type's reversal potential multiplied by a conductance:

$$h_{leak}(\mathbf{x};\mathbf{z}) = g_{leak}(x_\alpha - V_{leak}) \tag{41}$$

$$h_{elec}(\mathbf{x};\mathbf{z}) = g_{\text{el}}(x_\alpha^{post} - x_\alpha^{pre}) \tag{42}$$

$$h_{syn}(\mathbf{x};\mathbf{z}) = g_{syn}S_\infty^{pre}(x_\alpha^{post} - V_{syn}) \tag{43}$$

$$h_{Ca}(\mathbf{x};\mathbf{z}) = g_{Ca}M_\infty(x_\alpha - V_{Ca}) \tag{44}$$

$$h_K(\mathbf{x};\mathbf{z}) = g_K N(x_\alpha - V_K) \tag{45}$$

$$h_{hyp}(\mathbf{x};\mathbf{z}) = g_h H(x_\alpha - V_{hyp}). \tag{46}$$

The reversal potentials were set to $V_{leak} = -40mV$, $V_{Ca} = 100mV$, $V_K = -80mV$, $V_{hyp} = -20mV$, and $V_{syn} = -75mV$. The other conductance parameters were fixed to $g_{leak} = 1 \times 10^{-4}\mu S$. g_{Ca}, g_K, and g_{hyp} had different values based on fast, intermediate (hub) or slow neuron. The fast conductances had values $g_{Ca} = 1.9 \times 10^{-2}$, $g_K = 3.9 \times 10^{-2}$, and $g_{hyp} = 2.5 \times 10^{-2}$. The intermediate conductances had values $g_{Ca} = 1.7 \times 10^{-2}$, $g_K = 1.9 \times 10^{-2}$, and $g_{hyp} = 8.0 \times 10^{-3}$. Finally, the slow conductances had values $g_{Ca} = 8.5 \times 10^{-3}$, $g_K = 1.5 \times 10^{-2}$, and $g_{hyp} = 1.0 \times 10^{-2}$.

Furthermore, the Calcium, Potassium, and hyperpolarization channels have time-dependent gating dynamics dependent on steady-state gating variables M_∞, N_∞ and H_∞, respectively:

$$M_\infty = 0.5\left(1 + \tanh\left(\frac{x_\alpha - v_1}{v_2}\right)\right) \tag{47}$$

$$\frac{dN}{dt} = \lambda_N(N_\infty - N) \tag{48}$$

$$N_\infty = 0.5\left(1 + \tanh\left(\frac{x_\alpha - v_3}{v_4}\right)\right) \tag{49}$$

$$\lambda_N = \phi_N \cosh\left(\frac{x_\alpha - v_3}{2v_4}\right) \tag{50}$$

$$\frac{dH}{dt} = \frac{(H_\infty - H)}{\tau_h} \tag{51}$$

$$H_\infty = \frac{1}{1 + \exp\left(\frac{x_\alpha + v_5}{v_6}\right)} \tag{52}$$

$$\tau_h = 272 - \left(\frac{-1499}{1 + \exp\left(\frac{-x_\alpha + v_7}{v_8}\right)}\right). \tag{53}$$

where we set $v_1 = 0mV$, $v_2 = 20mV$, $v_3 = 0mV$, $v_4 = 15mV$, $v_5 = 78.3mV$, $v_6 = 10.5mV$, $v_7 = -42.2mV$, $v_8 = 87.3mV$, $v_9 = 5mV$, and $v_{th} = -25mV$.

Finally, there is a synaptic gating variable as well:

$$S_\infty = \frac{1}{1 + \exp\left(\frac{v_{th} - x_\alpha}{v_9}\right)}. \tag{54}$$

When the dynamic gating variables are considered, this is actually a 15-dimensional nonlinear dynamical system. Gaussian noise $d\mathbf{B}$ of variance $(1 \times 10^{-12})^2$ A^2 makes the model stochastic, and introduces variability in frequency at each parameterization $\mathbf{z}$.

In order to measure the frequency of the hub neuron during EPI, the STG model was simulated for $T = 300$ time steps of $dt = 25$ms. The chosen dt and T were the most computationally convenient choices yielding accurate frequency measurement. We used a basis of complex exponentials with frequencies from 0.0-1.0 Hz at 0.01Hz resolution to measure frequency from simulated time series

$$\Phi = [0.0, 0.01, ..., 1.0]^\top .. \tag{55}$$

To measure spiking frequency, we processed simulated membrane potentials with a relu (spike extraction) and low-pass filter with averaging window of size 20, then took the frequency with the maximum absolute value of the complex exponential basis coefficients of the processed time-series. The first 20 temporal samples of the simulation are ignored to account for initial transients.

To differentiate through the maximum frequency identification, we used a soft-argmax Let $X_\alpha \in \mathcal{C}^{|\Phi|}$ be the complex exponential filter bank dot products with the signal $x_\alpha \in \mathbb{R}^N$, where $\alpha \in \{\mathrm{f1}, \mathrm{f2}, \mathrm{hub}, \mathrm{s1}, \mathrm{s2}\}$. The soft-argmax is then calculated using temperature parameter $\beta = 100$

$$\psi_\alpha = \mathrm{softmax}(\beta |X_\alpha| \odot i), \tag{56}$$

where $i = [0, 1, ..., 100]$. The frequency is then calculated as

$$\omega_\alpha = 0.01 \psi_\alpha \mathrm{Hz}. \tag{57}$$

Intermediate hub frequency, like all other emergent properties in this work, is defined by the mean and variance of the emergent property statistics. In this case, we have one statistic, hub neuron frequency, where the mean was chosen to be 0.55Hz, and variance was chosen to be $(0.025\mathrm{Hz})^2$ to capture variation in frequency between 0.5Hz and 0.6Hz (Equation 4). As a maximum entropy distribution, $T(\mathbf{x}, \mathbf{z})$ is comprised of both these first and second moments of the hub neuron frequency

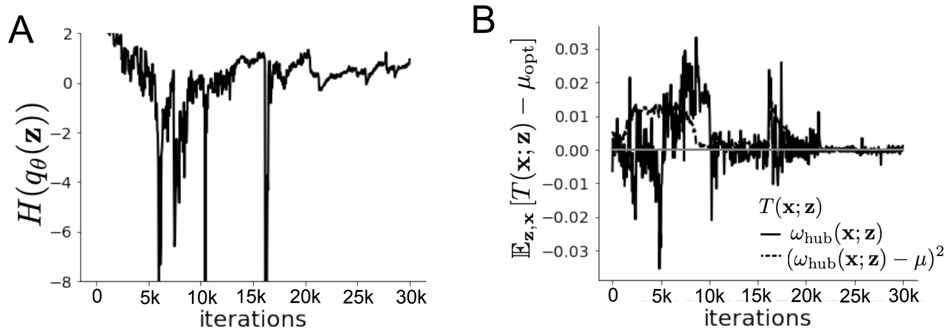

Figure 9: (STG1): EPI optimization of the STG model producing network syncing. **A**. Entropy throughout optimization. **B**. The emergent property statistic means and variances converge to their constraints at 25,000 iterations following the fifth augmented Lagrangian epoch.

(as in Equations 27 and 28)

$$T(\mathbf{x};\mathbf{z}) = \begin{bmatrix} \omega_{\text{hub}}(\mathbf{x};\mathbf{z}) \\ (\omega_{\text{hub}}(\mathbf{x};\mathbf{z}) - 0.55)^2 \end{bmatrix}, \tag{58}$$

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} 0.55 \\ 0.025^2 \end{bmatrix}. \tag{59}$$

Throughout optimization, the augmented Lagrangian parameters η and c, were updated after each epoch of 5,000 iterations(see Section 5.1.3). The optimization converged after five epochs (Fig. S4).

For EPI in Fig 1E, we used a real NVP architecture with three Real NVP coupling layers and two-layer neural networks of 25 units per layer. The normalizing flow architecture mapped $z_0 \sim \mathcal{N}(\mathbf{0}, I)$ to a support of $\mathbf{z} = [g_{\text{el}}, g_{\text{synA}}] \in [4, 8] \times [0.01, 4]$, initialized to a gaussian approximation of samples returned by a preliminary ABC search. We did not include $g_{\text{synA}} < 0.01$, for numerical stability. EPI optimization was run using 5 different random seeds for architecture initialization $\boldsymbol{\theta}$ with an augmented Lagrangian coefficient of $c_0 = 10^5$, a batch size $n = 400$, and $\beta = 2$. The distribution shown is that of the architecture converging with criteria $N_{\text{test}} = 100$ at greatest entropy across random seeds.

We calculated the Hessian at the mode of the inferred EPI distribution. The Hessian of a probability model is the second order gradient of the log probability density $\log q_{\boldsymbol{\theta}}(\mathbf{z})$ with respect to the parameters $\mathbf{z}$: $\frac{\partial^2 \log q_{\boldsymbol{\theta}}(\mathbf{z})}{\partial \mathbf{z} \partial \mathbf{z}^\top}$. With EPI, we can examine the Hessian, which is analytically available throughout distribution, to indicate the dimensions of parameter space that are sensitive (strongly negative eigenvalue), and which are degenerate (low magnitude eigenvalue) with respect to the emergent property produced. In Figure 1D, the eigenvectors of the Hessian v_1 (solid) and v_2 (dashed) are shown evaluated at the mode of the distribution. The length of the arrows is inversely proportional to the square root of absolute value of their eigenvalues $\lambda_1 = -10.7$ and $\lambda_2 = -3.22$. Since the Hessian eigenvectors have sign degeneracy, the visualized directions in 2-D parameter space are chosen arbitrarily.

5.2.2 Scaling EPI for stable amplification in RNNs

We examined the scaling properties of EPI by learning connectivities of RNNs of increasing size that exhibit stable amplification. Rank-2 RNN connectivity was modeled as $W = UV^\top$, where $U = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 \end{bmatrix} + g\chi^{(W)}$, $V = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{bmatrix} + g\chi^{(V)}$, and $\chi^{(W)}_{i,j}, \chi^{(V)}_{i,j} \sim \mathcal{N}(0,1)$. This RNN model has dynamics

$$\tau \dot{\mathbf{x}} = -\mathbf{x} + W\mathbf{x}. \tag{60}$$

In this analysis, we inferred connectivity parameterizations $\mathbf{z} = \left[\mathbf{u}_1^\top, \mathbf{u}_2^\top, \mathbf{v}_1^\top, \mathbf{v}_2^\top\right]^\top \in [-1,1]^{(4N)}$ that produced stable amplification using EPI, SMC-ABC [43], and SNPE [45] (see Section Related Methods).

For this RNN model to be stable, all real eigenvalues of W must be less than 1: $\text{real}(\lambda_1) < 1$, where λ_1 denotes the greatest real eigenvalue of W. For a stable RNN to amplify at least one input pattern, the symmetric connectivity $W^s = \frac{W + W\top}{2}$ must have an eigenvalue greater than 1: $\lambda_1^s > 1$, where λ^s is the maximum eigenvalue of W^s. These two conditions are necessary and sufficient for stable amplification in RNNs [63]. We defined the emergent property of stable amplification with means of these eigenvalues (0.5 and 1.5, respectively) that satisfy these conditions. To complete the emergent property definition, we chose variances (0.25^2) about those means such that samples rarely violate the eigenvalue constraints. In terms of the EPI optimization variables, this is written as

$$T(\mathbf{x};\mathbf{z}) = \begin{bmatrix} \text{real}(\lambda_1)(\mathbf{x};\mathbf{z}) \\ \lambda_1^s(\mathbf{x};\mathbf{z}) \\ (\text{real}(\lambda_1)(\mathbf{x};\mathbf{z}) - 0.5)^2 \\ (\lambda_1^s(\mathbf{x};\mathbf{z}) - 1.5)^2 \end{bmatrix}, \tag{61}$$

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} 0.5 \\ 1.5 \\ 0.25^2 \\ 0.25^2 \end{bmatrix}. \tag{62}$$

Gradients of maximum eigenvalues of Hermitian matrices like W^s are available with modern automatic differentiation tools. To differentiate through the real(λ_1), we solved the following equation for eigenvalues of rank-2 matrices using the rank reduced matrix $W^r = V^\top U$

$$\lambda_\pm = \frac{\text{Tr}(W^r) \pm \sqrt{\text{Tr}(W^r)^2 - 4\text{Det}(W^r)}}{2}. \tag{63}$$

For EPI in Fig. 2, we used a real NVP architecture with three coupling layers of affine transformations parameterized by two-layer neural networks of 100 units per layer. The initial distribution was a standard isotropic gaussian $z_0 \sim \mathcal{N}(\mathbf{0}, I)$ mapped to the support of $\mathbf{z}_i \in [-1, 1]$. We used an augmented Lagrangian coefficient of $c_0 = 10^3$, a batch size $n = 200$, $\beta = 4$, and chose to use 500 iterations per augmented Lagrangian epoch and emergent property constraint convergence was evaluated at $N_{\text{test}} = 200$ (Fig. 2B blue line, and Fig. 2C-D blue).

We compared EPI to two alternative simulation-based inference techniques, since the likelihood of these eigenvalues given $\mathbf{z}$ is not available. Approximate Bayesian computation (ABC) [80] is a rejection sampling technique for obtaining sets of parameters $\mathbf{z}$ that produce activity $\mathbf{x}$ close to some observed data $\mathbf{x}_0$. Sequential Monte Carlo approximate Bayesian computation (SMC-ABC) is the state-of-the-art ABC method, which leverages SMC techniques to improve sampling speed. We ran SMC-ABC with the pyABC package [105] to infer RNNs with stable amplification: connectivities having eigenvalues within an ϵ-defined l-2 distance of

$$x_0 = \begin{bmatrix} \text{real}(\lambda_1) \\ \lambda_1^s \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix}. \tag{64}$$

SMC-ABC was run with a uniform prior over $\mathbf{z} \in [-1, 1]^{(4N)}$, a population size of 1,000 particles with simulations parallelized over 32 cores, and a multivariate normal transition model.

SNPE, the next approach in our comparison, is far more similar to EPI. Like EPI, SNPE treats parameters in mechanistic models with deep probability distributions, yet the two learning algorithms are categorically different. SNPE uses a two-network architecture to approximate the posterior distribution of the model conditioned on observed data $\mathbf{x}_0$. The amortizing network maps observations $\mathbf{x}_i$ to the parameters of the deep probability distribution. The weights and biases of the parameter network are optimized by sequentially augmenting the training data with additional pairs ($\mathbf{z}_i$, $\mathbf{x}_i$) based on the most recent posterior approximation. This sequential procedure is important to get training data $\mathbf{z}_i$ to be closer to the true posterior, and $\mathbf{x}_i$ to be closer to the observed data. For the deep probability distribution architecture, we chose a masked autoregressive flow with affine couplings (the default choice), three transforms, 50 hidden units, and a normalizing flow mapping

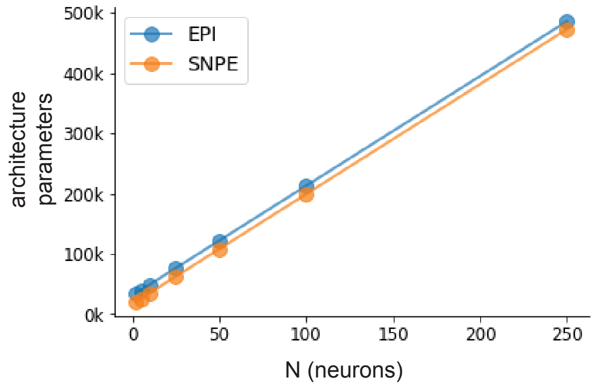

Figure 10: (RNN1): Number of parameters in deep probability distribution architectures of EPI (blue) and SNPE (orange) by RNN size (N).

to the support as in EPI. This architectural choice closely tracked the size of the architecture used by EPI (Fig. 10). As in SMC-ABC, we ran SNPE with $\mathbf{x}_0 = \mu$. All SNPE optimizations were run for a limit of 1.5 days on a Tesla V100 GPU, or until two consecutive rounds resulted in a validation log probability lower than the maximum observed for that random seed.

To clarify the difference in objectives of EPI and SNPE, we show their results on RNN models with different numbers of neurons N and random strength g. The parameters inferred by EPI consistently produces the same mean and variance of real(λ_1) and λ_1^s, while those inferred by SNPE change according to the model definition (Fig. 11A). For $N = 2$ and $g = 0.01$, the SNPE posterior has greater concentration in eigenvalues around $\mathbf{x}_0$ than at $g = 0.1$, where the model has greater randomness (Fig. 11B top, orange). At both levels of g when $N = 2$, the posterior of SNPE has lower entropy than EPI at convergence (Fig. 11B top). However at $N = 10$, SNPE results in a predictive distribution of more widely dispersed eigenvalues (Fig. 11A bottom), and an inferred posterior with greater entropy than EPI (Fig. 11B bottom). We highlight these differences not to focus on an insightful trend, but to emphasize that these methods optimize different objectives with different implications.

Note that SNPE converges when it's validation log probability has saturated after several rounds of optimization (Fig. 11C), and that EPI converges after several epochs of its own optimization to enforce the emergent property constraints (Fig. 11D blue). Importantly, as SNPE optimizes its posterior approximation, the predictive means change, and at convergence may be different than $\mathbf{x}_0$ (Fig. 11D orange, left). It is sensible to assume that predictions of a well-approximated SNPE posterior should closely reflect the data on average (especially given a uniform prior and

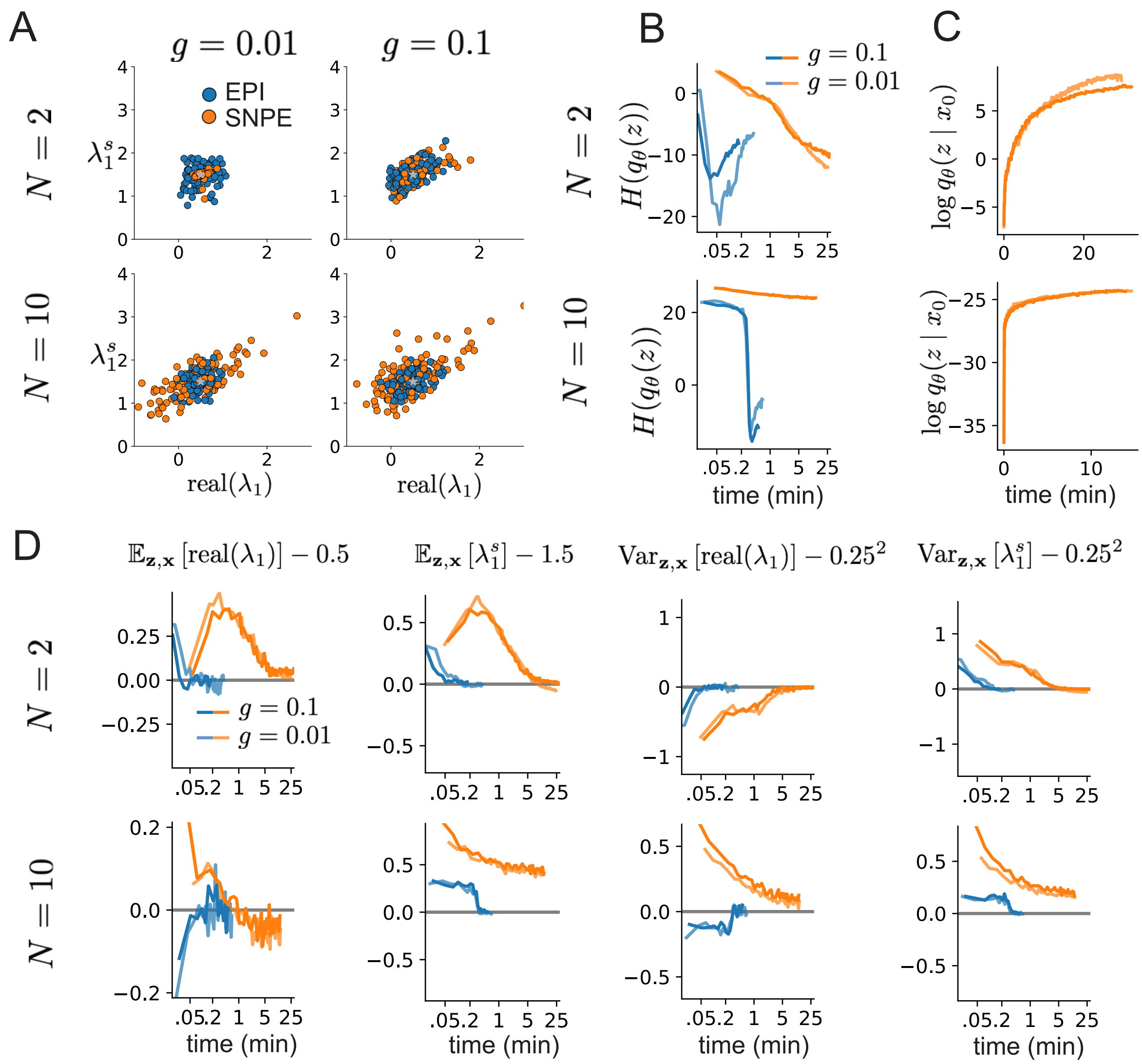

Figure 11: (RNN2): Model characteristics affect predictions of posteriors inferred by SNPE, while predictions of parameters inferred by EPI remain fixed. **A**. Predictive distribution of EPI (blue) and SNPE (orange) inferred connectivity of RNNs exhibiting stable amplification with $N = 2$ (top), $N = 10$ (bottom), $g = 0.01$ (left), and $g = 0.1$ (right). **B**. Entropy of parameter distribution approximations throughout optimization with $N = 2$ (top), $N = 10$ (bottom), $g = 0.1$ (dark shade), and $g = 0.01$ (light shade). **C**. Validation log probabilities throughout SNPE optimization. Same conventions as B. **D**. Adherence to EPI constraints. Same conventions as B.

a low degree of stochasticity), however this is not a given. Furthermore, no aspect of the SNPE optimization controls the variance of the predictions (Fig. 11D orange, right).

To compare the efficiency of these algorithms for inferring RNN connectivity distributions producing stable amplification, we develop a convergence criteria that can be used across methods. While EPI has its own hypothesis testing convergence criteria for the emergent property, it would not make sense to use this criteria on SNPE and SMC-ABC which do not constrain the means and variances of their predictions. Instead, we consider EPI and SNPE to have converged after completing its most recent optimization epoch (EPI) or round (SNPE) in which the distance

$$d(q_\theta(z)) = |\mathbb{E}_{\mathbf{z},\mathbf{x}}\left[f(\mathbf{x};\mathbf{z})\right] - \boldsymbol{\mu}|_2 \tag{65}$$

is less than 0.5. We consider SMC-ABC to have converged once the population produces samples within the $\epsilon = 0.5$ ball ensuring stable amplification.

When assessing the scalability of SNPE, it is important to check that alternative hyperparamterizations could not yield better performance. Key hyperparameters of the SNPE optimization are the number of simulations per round n_{round}, the number of atoms used in the atomic proposals of the SNPE-C algorithm [106], and the batch size n. To match EPI, we used a batch size of $n = 200$ for $N <= 25$, however we found $n = 1,000$ to be helpful for SNPE in higher dimensions. While $n_{\text{round}} = 1,000$ yielded SNPE convergence for $N <= 25$, we found that a substantial increase to $n_{\text{round}} = 25,000$ yielded more consistent convergence at $N = 50$ (Fig. 12A). By increasing n_{round}, we also necessarily increase the duration of each round. At $N = 100$, we tried two hyperparameter modifications. As suggested in [106], we increased n_{atom} by an order of magnitude to improve gradient quality, but this had little effect on the optimization (much overlap between same random seeds) (Fig. 12B). Finally, we increased n_{round} by an order of magnitude, which yielded convergence in one case, but no others. We found no way to improve the convergence rate of SNPE without making more aggressive hyperparameter choices requiring high numbers of simulations.

In Figure 2C-D, we show samples form the random seed resulting in emergent property convergence at greatest entropy (EPI), the random seed resulting in greatest validation log probability (SNPE), and the result of all converged random seeds (SMC).

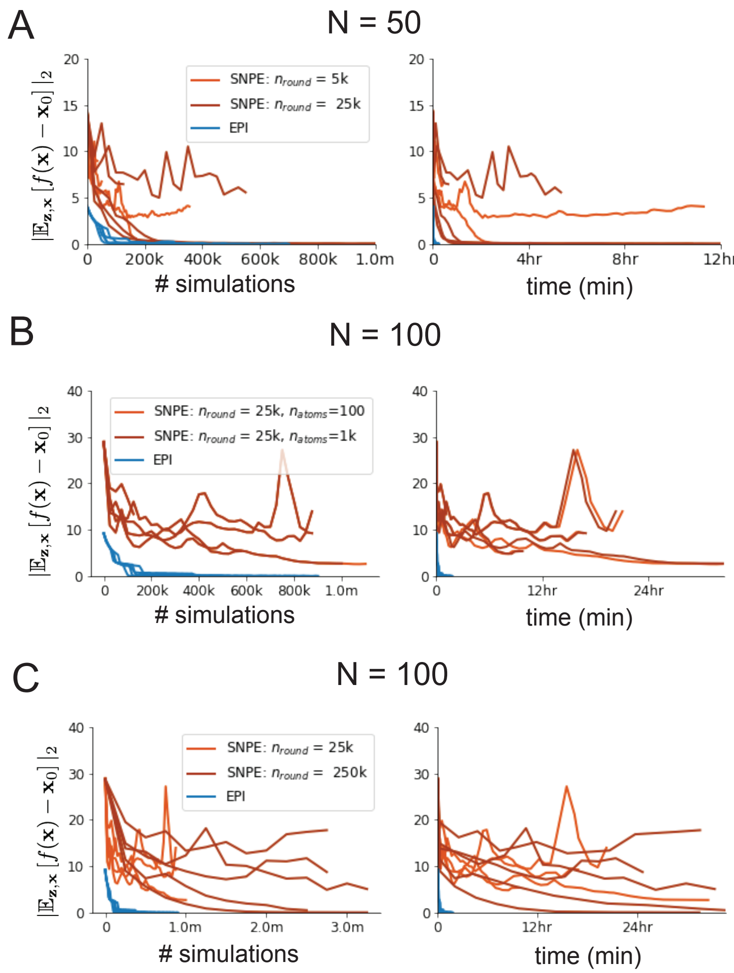

Figure 12: (RNN3): SNPE convergence was enabled by increasing n_{round}, not n_{atom}. **A**. Difference of mean predictions $\mathbf{x}_0$ throughout optimization at $N = 50$ with by simulation count (left) and wall time (right) of SNPE with $n_{\text{round}} = 5,000$ (light orange), SNPE with $n_{\text{round}} = 25,000$ (dark orange), and EPI (blue). Each line shows an individual random seed. **B**. Same conventions as A at $N = 100$ of SNPE with $n_{\text{atom}} = 100$ (light orange) and $n_{\text{atom}} = 1,000$ (dark orange). **C**. Same conventions as A at $N = 100$ of SNPE with $n_{\text{round}} = 25,000$ (light orange) and $n_{\text{round}} = 250,000$ (dark orange).

5.2.3 Primary visual cortex

In the stochastic stabilized supralinear network [78], population rate responses $\mathbf{x}$ to input $\mathbf{h}$, recurrent input $W\mathbf{x}$ and slow noise $\boldsymbol{\epsilon}$ are governed by

$$\tau \frac{d\mathbf{x}}{dt} = -\mathbf{x} + \phi(W\mathbf{x} + \mathbf{h} + \boldsymbol{\epsilon}), \tag{66}$$

where the noise is an Ornstein-Uhlenbeck process $\boldsymbol{\epsilon} \sim OU(\tau_{\text{noise}}, \boldsymbol{\sigma})$

$$\tau_{\text{noise}} d\epsilon_\alpha = -\epsilon_\alpha dt + \sqrt{2\tau_{\text{noise}}}\tilde{\sigma}_\alpha dB \tag{67}$$

with $\tau_{\text{noise}} = 5\text{ms} > \tau = 1\text{ms}$. The noisy process is parameterized as

$$\tilde{\sigma}_\alpha = \sigma_\alpha \sqrt{1 + \frac{\tau}{\tau_{\text{noise}}}}, \tag{68}$$

so that $\boldsymbol{\sigma}$ parameterizes the variance of the noisy input in the abscence of recurrent connectivity ($W = \mathbf{0}$). As contrast increases, input to the E- and P-populations increases relative to a baseline input $\mathbf{h} = \mathbf{h}_b + c\mathbf{h}_c$. Connectivity ($W_{\text{fit}}$) and input ($\mathbf{h}_{b,\text{fit}}$ and $\mathbf{h}_{c,\text{fit}}$) parameters were fit using the deterministic V1 circuit model [56]

$$W_{\text{fit}} = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & W_{EV} \\ W_{PE} & W_{PP} & W_{PS} & W_{PV} \\ W_{SE} & W_{SP} & W_{SS} & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & W_{VV} \end{bmatrix} = \begin{bmatrix} 2.18 & -1.19 & -.594 & -.229 \\ 1.66 & -.651 & -.680 & -.242 \\ .895 & -5.22 \times 10^{-3} & -1.51 \times 10^{-4} & -.761 \\ 3.34 & -2.31 & -.254 & -2.52 \times 10^{-4} \end{bmatrix}, \tag{69}$$

$$\mathbf{h}_{b,\text{fit}} = \begin{bmatrix} .416 \\ .429 \\ .491 \\ .486 \end{bmatrix}, \tag{70}$$

and

$$\mathbf{h}_{c,\text{fit}} = \begin{bmatrix} .359 \\ .403 \\ 0 \\ 0 \end{bmatrix}. \tag{71}$$

To obtain rates on a realistic scale (100-fold greater), we map these fitted parameters to an equivalence class

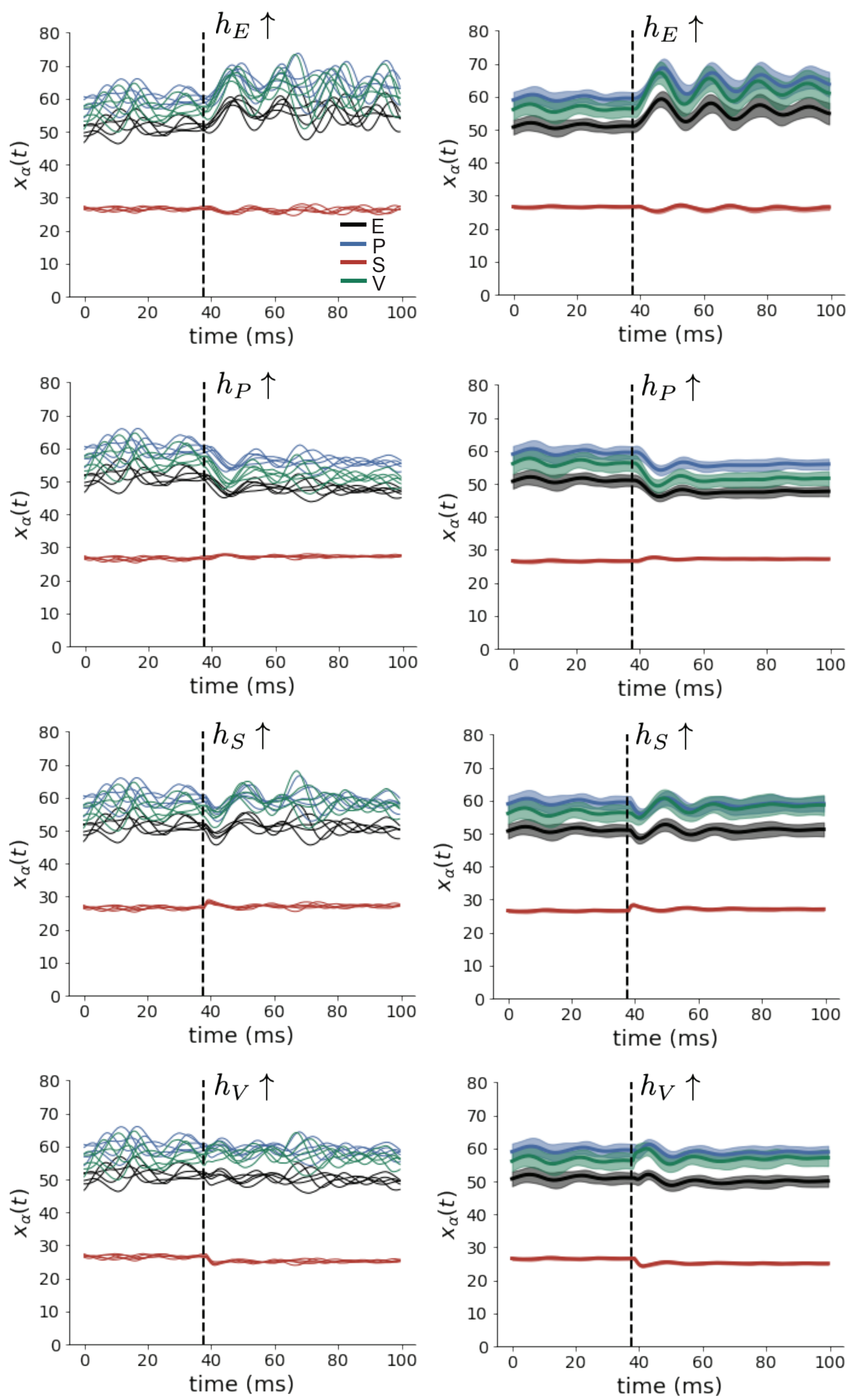

Figure 13: (V1 1) (Left) Simulations for small increases in neuron-type population input. Input magnitudes are chosen so that effect is salient (0.002 for E and P, but 0.02 for S and V). (Right) Average (solid) and standard deviation (shaded) of stochastic fluctuations of responses.

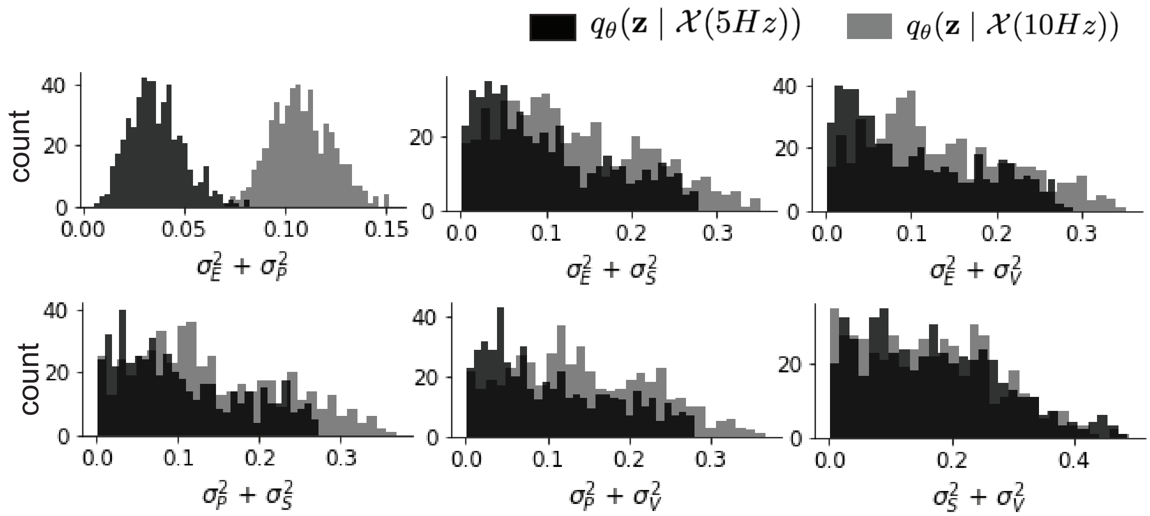

Figure 14: (V1 2) EPI predictive distributions of the sum of squares of each pair of noise parameters.

$$W = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & W_{EV} \\ W_{PE} & W_{PP} & W_{PS} & W_{PV} \\ W_{SE} & W_{SP} & W_{SS} & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & W_{VV} \end{bmatrix} = \begin{bmatrix} .218 & -.119 & -.0594 & -.0229 \\ .166 & -.0651 & -.068 & -.0242 \\ .0895 & -5.22 \times 10^{-4} & -1.51 \times 10^{-5} & -.0761 \\ .334 & -.231 & -.0254 & -2.52 \times 10^{-5} \end{bmatrix}, \tag{72}$$

$$\mathbf{h}_b = \begin{bmatrix} h_{b,E} \\ h_{b,P} \\ h_{b,S} \\ h_{b,V} \end{bmatrix} = \begin{bmatrix} 4.16 \\ 4.29 \\ 4.91 \\ 4.86 \end{bmatrix}, \tag{73}$$

1224 and

$$\mathbf{h}_c = \begin{bmatrix} h_{c,E} \\ h_{c,P} \\ h_{c,S} \\ h_{c,V} \end{bmatrix} = \begin{bmatrix} 3.59 \\ 4.03 \\ 0 \\ 0 \end{bmatrix}. \tag{74}$$

1225 Circuit responses are simulated using $T = 200$ time steps at $dt = 0.5$ms from an initial condition
1226 drawn from $\mathbf{x}(0) \sim U\,[10 \text{ Hz}, 25 \text{ Hz}]$. Standard deviation of the E-population $s_E(\mathbf{x};\mathbf{z})$ is calculated
1227 as the square root of the temporal variance from $t_{ss} = 75$ms to $Tdt = 100$ms averaged over 100

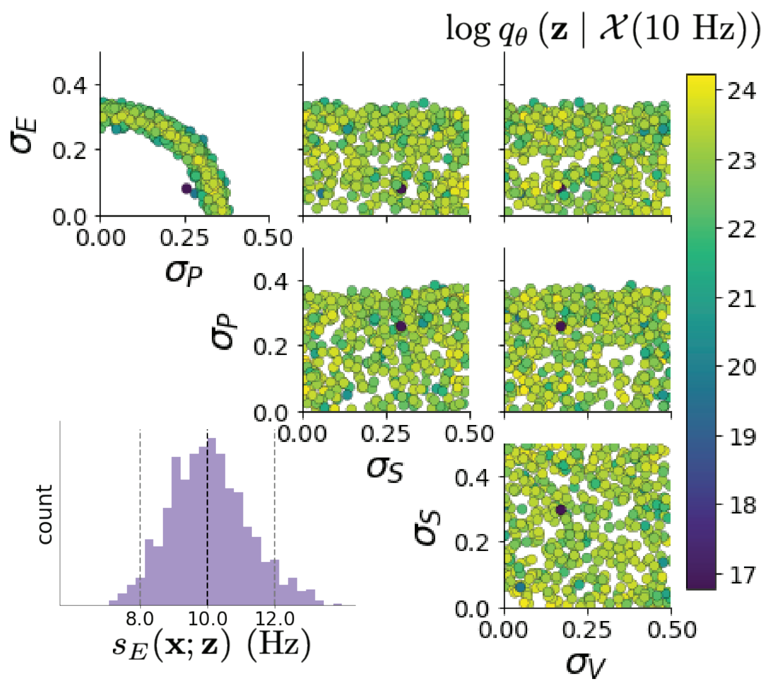

Figure 15: (V1 3) EPI inferred distribution for $\mathcal{X}(10\text{ Hz})$.

independent trials.

$$s_E(\mathbf{x};\mathbf{z}) = \mathbb{E}_x\left[\sqrt{\mathbb{E}_{t>t_{ss}}\left[\left(x_E(t) - \mathbb{E}_{t>t_{ss}}\left[x_E(t)\right]\right)^2\right]}\right] \tag{75}$$

For EPI in Fig 3D-E, we used a real NVP architecture with three Real NVP coupling layers and two-layer neural networks of 50 units per layer. The normalizing flow architecture mapped $z_0 \sim \mathcal{N}(\mathbf{0}, I)$ to a support of $\mathbf{z} = [\sigma_E, \sigma_P, \sigma_S, \sigma_V] \in [0.0, 0.5]^4$. EPI optimization was run using three different random seeds for architecture initialization $\boldsymbol{\theta}$ with an augmented Lagrangian coefficient of $c_0 = 10^{-1}$, a batch size $n = 100$, and $\beta = 2$. The distributions shown are those of the architectures converging with criteria $N_{\text{test}} = 100$ at greatest entropy across random seeds.

In Fig. 3E, we visualize the modes of $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$ throughout the σ_E-σ_P marginal. Specifically, we calculated

$$\begin{aligned} \mathbf{z}^*(\sigma_{P,\text{fixed}}) = &\underset{\mathbf{z}}{\operatorname{argmax}} \log q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X}) \\ &\text{s.t. } \sigma_P = \sigma_{P,\text{fixed}} \end{aligned} \tag{76}$$

At each mode $\mathbf{z}^*$, we calculated the Hessian and visualized the sensitivity dimension in the direction of positive σ_E.

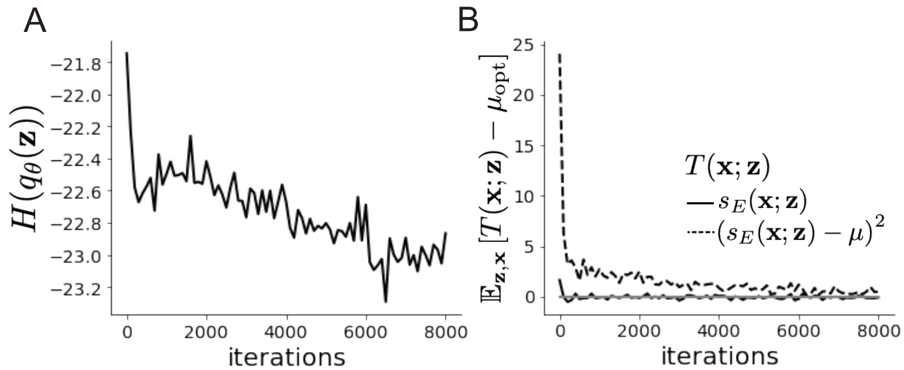

Figure 16: (V1 4) Optimization for V1

5.2.4 Primary visual cortex: challenges to analysis

TODO Agostina and I are putting this together now.

5.2.5 Superior colliculus

The ability to switch between two separate tasks throughout randomly interleaved trials, or "rapid task switching," has been studied in rats, and midbrain superior colliculus (SC) has been show to play an important in this computation [79]. Neural recordings in SC exhibited two populations of neurons that simultaneously represented both task context (Pro or Anti) and motor response (contralateral or ipsilateral to the recorded side), which led to the distinction of two functional classes: the Pro/Contra and Anti/Ipsi neurons [57]. Given this evidence, Duan et al. proposed a model with four functionally-defined neuron-type populations: two in each hemisphere corresponding to the Pro/Contra and Anti/Ipsi populations. We study how the connectivity of this neural circuit governs rapid task switching ability.

The four populations of this model are denoted as left Pro (LP), left Anti (LA), right Pro (RP) and right Anti (RA). Each unit has an activity (x_α) and internal variable (u_α) related by

$$x_\alpha = \phi(u_\alpha) = \left(\frac{1}{2}\tanh\left(\frac{u_\alpha - a}{b}\right) + \frac{1}{2}\right), \tag{77}$$

where $\alpha \in \{LP, LA, RA, RP\}$, $a = 0.05$ and $b = 0.5$ control the position and shape of the nonlinearity. We order the neural populations of x and u in the following manner

$$\mathbf{x} = \begin{bmatrix} x_{LP} \\ x_{LA} \\ x_{RP} \\ x_{RA} \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} u_{LP} \\ u_{LA} \\ u_{RP} \\ u_{RA} \end{bmatrix}, \tag{78}$$

which evolve according to

$$\tau \frac{d\mathbf{u}}{dt} = -\mathbf{u} + W\mathbf{x} + \mathbf{h} + d\mathbf{B}. \tag{79}$$

with time constant $\tau = 0.09s$, step size 24ms and Gaussian noise $d\mathbf{B}$ of variance 0.2^2. These hyperparameter values are motivated by modeling choices and results from [57].

The weight matrix has 4 parameters for self sW, vertical vW, horizontal hW, and diagonal dW connections:

$$W = \begin{bmatrix} sW & vW & hW & dW \\ vW & sW & dW & hW \\ hW & dW & sW & vW \\ dW & hW & vW & sW \end{bmatrix}. \tag{80}$$

We study the role of parameters $\mathbf{z} = [sW, vW, hW, dW]^\top$ in rapid task switching.

The circuit receives four different inputs throughout each trial, which has a total length of 1.8s.

$$\mathbf{h} = \mathbf{h}_{\text{constant}} + \mathbf{h}_{\text{P,bias}} + \mathbf{h}_{\text{rule}} + \mathbf{h}_{\text{choice-period}} + \mathbf{h}_{\text{light}}. \tag{81}$$

There is a constant input to every population,

$$\mathbf{h}_{\text{constant}} = I_{\text{constant}}[1, 1, 1, 1]\top, \tag{82}$$

a bias to the Pro populations

$$\mathbf{h}_{\text{P,bias}} = I_{\text{P,bias}}[1, 0, 1, 0]\top, \tag{83}$$

rule-based input depending on the condition

$$\mathbf{h}_{\text{P,rule}}(t) = \begin{cases} I_{\text{P,rule}}[1, 0, 1, 0]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \tag{84}$$

$$\mathbf{h}_{\text{A,rule}}(t) = \begin{cases} I_{\text{A,rule}}[0, 1, 0, 1]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases}, \tag{85}$$

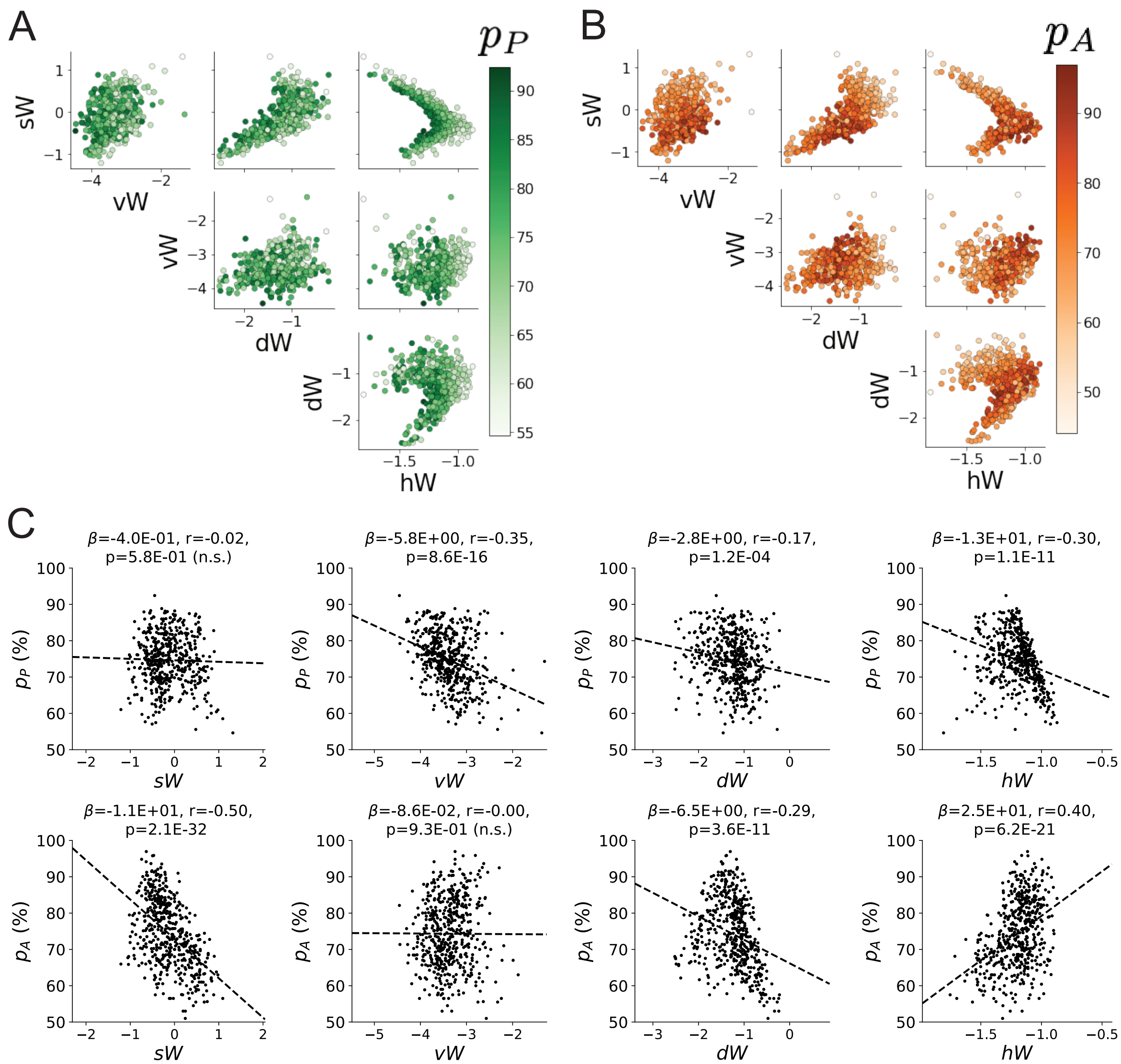

Figure 17: (SC1): **A**. Same pairplot as Fig. 4C colored by Pro task accuracy. **B**. Same as A colored by Anti task accuracy. **C**. Connectivity parameters of EPI distributions versus task accuracies. β is slope coefficient of linear regression, r is correlation, and p is the two-tailed p-value.

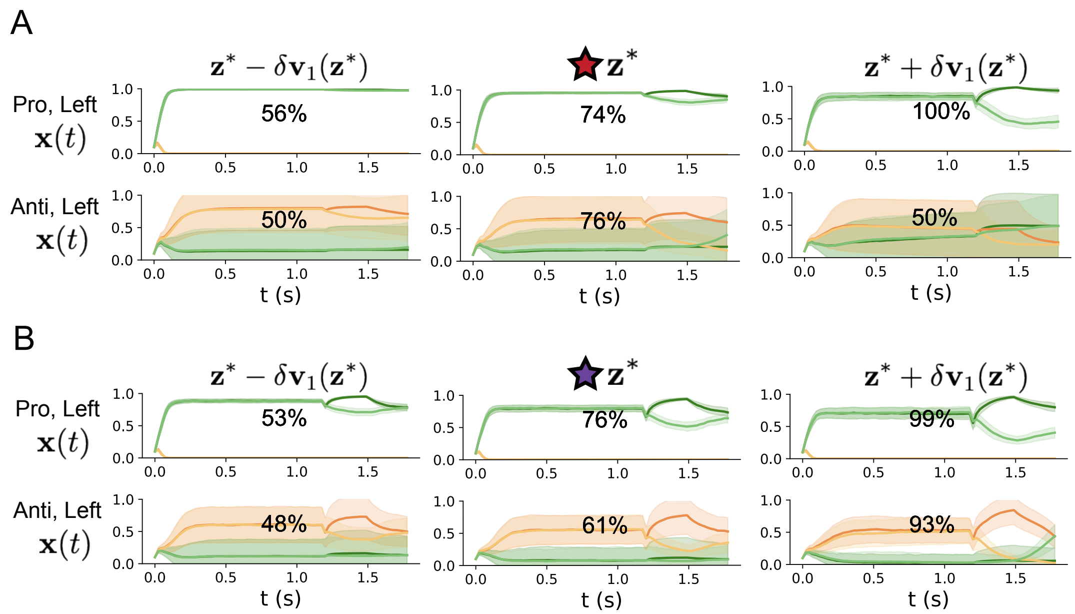

Figure 18: (SC2): **A**. Simulations in network regime 1 ($hW_{\text{fixed}} = -1.2$) (center) with simulations given connectivity perturbations in the negative direction of the sensitivity vector $\mathbf{v}_1$ (left) and positive direction (right). **B**. Same as A for network regime 2.

a choice-period input

$$\mathbf{h}_{\text{choice}}(t) = \begin{cases} I_{\text{choice}}[1,1,1,1]^\top, & \text{if } t > 1.2s \\ 0, & \text{otherwise} \end{cases}, \tag{86}$$

and an input to the right or left-side depending on where the light stimulus is delivered

$$\mathbf{h}_{\text{light}}(t) = \begin{cases} I_{\text{light}}[1,1,0,0]^\top, & \text{if } 1.2s < t < 1.5s \text{ and Left} \\ I_{\text{light}}[0,0,1,1]^\top, & \text{if } 1.2s < t < 1.5s \text{ and Right} \\ 0, & \text{otherwise} \end{cases}. \tag{87}$$

The input parameterization was fixed to $I_{\text{constant}} = 0.75$, $I_{\text{P,bias}} = 0.5$, $I_{\text{P,rule}} = 0.6$, $I_{\text{A,rule}} = 0.6$, $I_{\text{choice}} = 0.25$, and $I_{\text{light}} = 0.5$.

The accuracies of each task p_P and p_A are calculated as

$$p_P(\mathbf{x}; \mathbf{z}) = \mathbb{E}_{\mathbf{x}}\left[\Theta[x_{LP}(t = 1.8s) - x_{RP}(t = 1.8s)]\right] \tag{88}$$

and

$$p_A(\mathbf{x}; \mathbf{z}) = \mathbb{E}_{\mathbf{x}}\left[\Theta[x_{RP}(t = 1.8s) - x_{LP}(t = 1.8s)]\right] \tag{89}$$

given that the stimulus is on the left side, where Θ is the Heaviside step function, and the accuracy

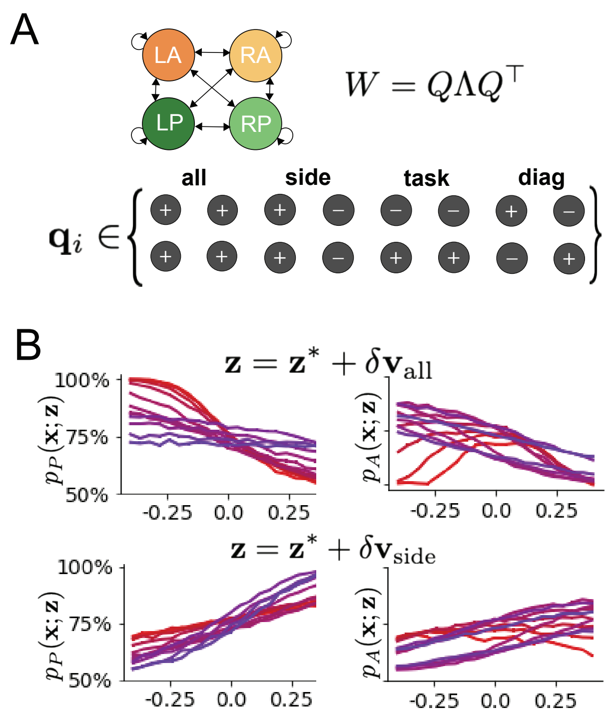

Figure 19: (SC3): **A**. Invariant eigenvectors of connectivity matrix W. **B**. Accuracies for connectivity perturbations for increasing λ_{all} and λ_{side} (rest shown in Fig. 4D).

is averaged over 200 independent trials. The Heaviside step function is approximated as

$$\Theta(\mathbf{x}) = \text{sigmoid}(\beta \mathbf{x}), \tag{90}$$

where $\beta = 100$.

Writing the EPI distribution as a maximum entropy distribution, $T(\mathbf{x}, \mathbf{z})$ is comprised of both these first and second moments of the accuracy in each task (as in Equations 27 and 28)

$$T(\mathbf{x}; \mathbf{z}) = \begin{bmatrix} p_P(\mathbf{x}; \mathbf{z}) \\ p_A(\mathbf{x}; \mathbf{z}) \\ (p_P(\mathbf{x}; \mathbf{z}) - 75\%)^2 \\ (p_A(\mathbf{x}; \mathbf{z}) - 75\%)^2 \end{bmatrix}, \tag{91}$$

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} 75\% \\ 75\% \\ 7.5\%^2 \\ 7.5\%^2 \end{bmatrix}. \tag{92}$$

Throughout optimization, the augmented Lagrangian parameters η and c, were updated after each

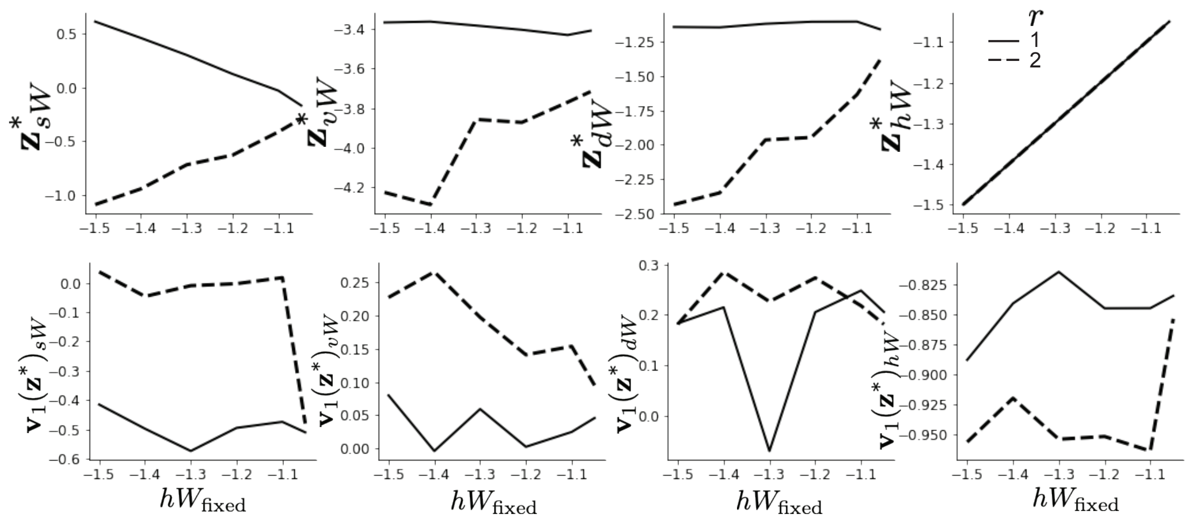

Figure 20: (SC4): **A**. The individual parameters of each mode throughout the two regimes. **B**. The individual sensitivities of parameters of each mode throughout the two regimes.

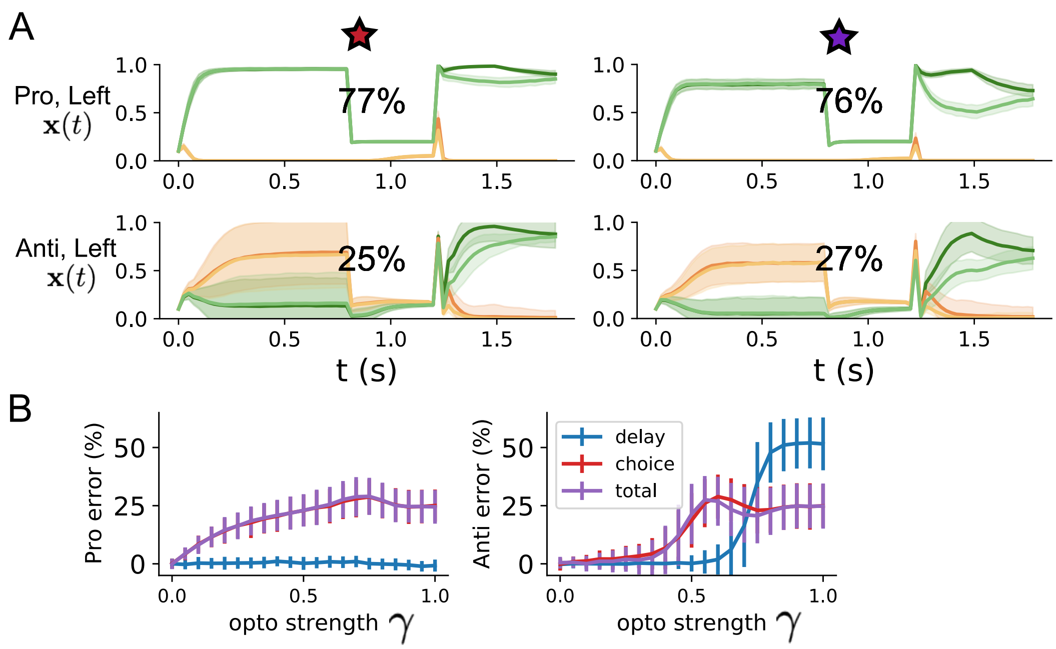

Figure 21: (SC5): **A**. Response of each parameter regime to optogenetic silencing during the delay period. **B**. Error induced by delay period inactivation with increasing optogenetic strength. Means and standard deviations are calculated across the entire EPI distribution.

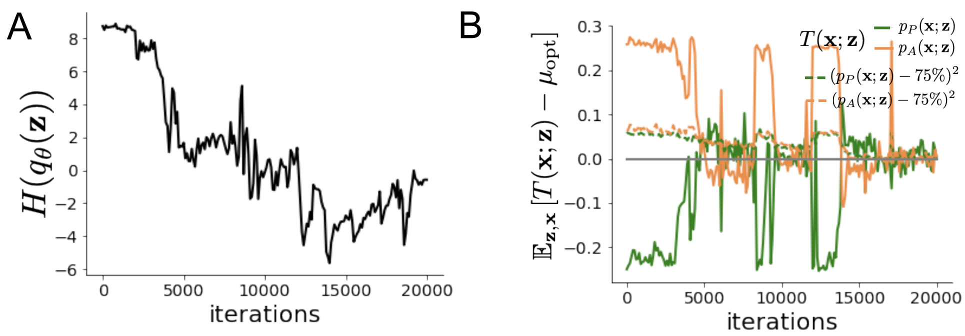

Figure 22: (SC6): **A**. Entropy throughout optimization. **B**. The emergent property statistic means and variances converge to their constraints at 20,000 iterations following the tenth augmented Lagrangian epoch.

epoch of 2,000 iterations (see Section 5.1.3). The optimization converged after ten epochs (Fig. 22).

For EPI in Fig. 4C, we used a real NVP architecture with three coupling layers of affine transformations parameterized by two-layer neural networks of 50 units per layer. The initial distribution was a standard isotropic gaussian $z_0 \sim \mathcal{N}(\mathbf{0}, I)$ mapped to a support of $\mathbf{z}_i \in [-5, 5]$. We used an augmented Lagrangian coefficient of $c_0 = 10^2$, a batch size $n = 100$, and $\beta = 2$. The distribution converged with criteria $N_{\text{test}} = 25$.

The EPI distribution of SC model connectivities producing rapid task switching has interesting structure. Throughout $q_{\boldsymbol{theta}}(\mathbf{z} \mid \mathcal{X})$, we see that the probability distribution is narrow in hW (Fig. 4C). This suggests that rapid task switching is sensitive to changes in hW, but this is only a single parameter. The local structure of the distribution varies across parameter space, and thus the nature in which parameter combinations affect rapid task switching. From visual inspection, we may hypothesize that there are two distinct regimes, most easily visualized in the sW-hW marginal distribution: one where sW and hW are correlated for greater sW and one where sW and hW are anticorrelated for lesser sW.

We sought two sets of parameters in this distribution representative of each regime, so that we could assess their implications on computation. For fixed values of hW, we hypothesized that there are two modes: one in each regime of greater and lesser sW. To begin, we found one mode for each regime at $hW_{\text{fixed}} = -1.5$ using 200 steps of gradient ascent of the deep probability distribution $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathcal{X})$. In regime 1, the initialization had positive sW, and the initialization had negative sW in regime 2, which led to disparate modes (Fig. 20 top). These modes were then used as the initialization to find the next mode at $hW_{\text{fixed}} = -1.4$ and so on. 200 steps of gradient ascent were always taken, and learning rates of $2.5 \times 10^{-}4$ and $5 \times 10^{-}4$ were used for regimes 1 and 2, respectively. Each of these modes is denoted $\mathbf{z}^*(hW_{\text{fixed}}, r)$ for regime $r \in \{1, 2\}$.

At each mode, we measure the sensitivity dimension (that of most negative eigenvalue in the Hessian of the EPI distribution) $\mathbf{v}_1(\mathbf{z}^*)$. To resolve sign degeneracy in eigenvectors, we chose $\mathbf{v}_1(\mathbf{z}^*)$ to have negative element in hW. This tells us what parameter combination rapid task switching is most sensitive to at this parameter choice in the regime. We see that while the modes of each regime gradually converge to similar connectivities at $hW_{\text{fixed}} = -1.05$ (Fig. 20 top), the sensitivity dimensions remain categorically different throughout the two regimes (Fig. 20 bottom) Only at $hW_{\text{fixed}} = -1.05$ is there a flip in sensitivity from regime 2 to regime 1 (in $\mathbf{v}_1(\mathbf{z}^*)_{sW}$ and $\mathbf{v}_1(\mathbf{z}^*)_{hW}$). There is thus some ambiguity regarding the "regime" of $\mathbf{z}^*(-1.05, 2)$, since the mode is derived from an initialization in regime 2, but has sensitivity like regime 1. We can consider this as an intermediate transitionary region of parameter space between the two regimes. To emphasize this, $\mathbf{z}^*(-1.05, 1)$ and $\mathbf{z}^*(-1.05, 2)$ have the same color.

To understand the connectivity mechanisms governing task accuracy, we took the eigendecomposition of the symmetric connectivity matrices $W = Q \Lambda Q^{-1}$, which results in the same basis vectors $\mathbf{q}_i$ for all W parameterized by $\mathbf{z}$ (Fig. 19A). These basis vectors have intuitive roles in processing for this task, and are accordingly named the *all* eigenmode - all neurons co-fluctuate, *side* eigenmode - one side dominates the other, *task* eigenmode - the Pro or Anti populations dominate the other, and *diag* mode - Pro- and Anti-populations of opposite hemispheres dominate the opposite pair. Due to the parametric structure of the connectivity matrix, the parameters $\mathbf{z}$ are a linear function of the eigenvalues $\boldsymbol{\lambda} = [\lambda_{\text{all}}, \lambda_{\text{side}}, \lambda_{\text{task}} \lambda_{\text{diag}}]^\top$ associated with these eigenmodes.

$$\mathbf{z} = A \boldsymbol{\lambda} \tag{93}$$

$$A = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix}. \tag{94}$$

We are interested in the effect of raising or lowering the amplification of each eigenmode in the connectivity matrix. To test this, we calculate the unit vector of changes in the connectivity $\mathbf{z}$ that result from a change in the associated eigenvalues

$$\mathbf{v}_{\text{a}} = \frac{\frac{\partial \mathbf{z}}{\partial \lambda_{\text{a}}}}{|\frac{\partial \mathbf{z}}{\partial \lambda_{\text{a}}}|_2}, \tag{95}$$

where

$$\frac{\partial \mathbf{z}}{\partial \lambda_{\mathrm{a}}} = A\mathbf{e}_{\mathrm{a}}, \tag{96}$$

and e.g. $\mathbf{e}_{\mathrm{all}} = [1, 0, 0, 0]^\top$. So $\mathbf{v}_{\mathrm{a}}$ is the normalized column of A corresponding to eigenmode a. While perturbations in the sensitivity dimension $\mathbf{v}_1(\mathbf{z}^*)$ adapt with the mode $\mathbf{z}^*$ chosen, perturbations in $\mathbf{v}_a$ for $a \in \{\mathrm{all}, \mathrm{side}, \mathrm{text}, \mathrm{diag}\}$ are invariant to $\mathbf{z}$ (Equation 96).

We tested whether the inferred SC model connectivities could reproduce experimental effects of optogenetic inactivation in rats [79]. During periods of simulated optogenetic inactivation, activity was decreased proportional to the optogenetic strength γ

$$x_\alpha = (1 - \gamma)\phi(u_\alpha). \tag{97}$$

Delay period inactivation was from $0.8 < t < 1.2$, choice period inactivation was for $t > 1.2$ and total inactivation was for the entire trial.