Neuronal circuits for robust online fixed-point detection

Summary. A fundamental problem in systems neuroscience is understanding how the brain learns the non-linear dynamics of the complex world and identifies the environment's state. Data-driven learning in such high-dimensional spaces requires lifting the curse of dimensionality. One way of reducing dimensionality relies on extracting from observed trajectories the underlying topological skeleton, i.e., *fixed points* connected by *invariant manifolds*. Thus, *online* extraction of fixed points from input trajectories is an important task. Whereas Dynamic Mode Decomposition (DMD) provides a framework to calculate fixed points offline, efficiently extracting fixed points *online* remains an unsolved problem. Moreover, implementing the online algorithm in a biological neuronal circuit requires it to satisfy more constraints, such as local update rules and no reliance on external memory.

In this work, we propose two *biologically plausible* neural networks with multi-compartment neurons for *online* fixed-point detection. The first neuronal circuit (*circuit A*) employs *mostly local* learning rules to update synaptic weights and estimate the linearized forward dynamics with high accuracy, and then utilizes the learned recurrent circuit to infer nearby fixed points. The second algorithm (*circuit B*) is a simpler recurrent neural network with synaptic weights learned by anti-Hebbian plasticity. Experiments show that *circuit A* can *efficiently* and *robustly* detect stable and unstable fixed points and all saddle points in switch-linear and non-linear systems. Though *circuit B* satisfies biological constraints more strictly, it converges slowly in practice. Our circuits are potential building blocks for a larger neuronal circuit for systems identification and model-based control.

Problem formulation. Considering a time-invariant nonlinear dynamical system, $\mathbf{x}_t = \mathbf{F}(\mathbf{x}_{t-1}) + \boldsymbol{\epsilon}(t)$, with Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$, given a trajectory, our goal is to find a fixed point $\mathbf{b}$ s.t. $\mathbf{F}(\mathbf{b}) = \mathbf{b}$. Assuming $\mathbf{F}$ is differentiable, linearization around a fixed point $\mathbf{b}$ gives $\mathbf{x}_t \approx \mathbf{b} + \tilde{\mathbf{A}}(\mathbf{x}_{t-1} - \mathbf{b}) + \tilde{\boldsymbol{\epsilon}}_t$, where $\tilde{\mathbf{A}} := \mathbf{J}_{\mathbf{F}}(\mathbf{b})$ is the Jacobian of $\mathbf{F}$ at the fixed point. The problem becomes to detect fixed points $\mathbf{b}$ given the observation of a trajectory. The linear approximation is good near fixed points for a period of time because trajectories change slowly. Thus minimizing the linear approximation error provides a means to estimate fixed points.

Neuronal *circuit A*. Similar to dynamical mode decomposition (DMD)[1], we define the "future" and the "past" trajectories of length T, $\mathbf{X}_f = (\mathbf{x}_1, \cdots, \mathbf{x}_T)$ and $\mathbf{X}_p = (\mathbf{x}_0, \cdots, \mathbf{x}_{T-1})$ and derive an unbiased estimate $\tilde{\mathbf{A}} \approx \mathbf{C}_{fp}\mathbf{C}_{pp}^{-1}$, where $\mathbf{C}_{fp} := \frac{1}{T}\left(\mathbf{X}_f - \bar{\mathbf{X}}_f\right)\left(\mathbf{X}_p - \bar{\mathbf{X}}_p\right)^\top$ and $\mathbf{C}_{pp} := \frac{1}{T}\left(\mathbf{X}_p - \bar{\mathbf{X}}_p\right)\left(\mathbf{X}_p - \bar{\mathbf{X}}_p\right)^\top$ are the covariance matrices. In the offline setting, we can show that the least-square estimate of $\mathbf{b}$ is given by $\mathbf{b} \approx \left(\mathbf{I} - \mathbf{C}_{fp}\mathbf{C}_{pp}^{-1}\right)^+ \left(\bar{\mathbf{x}}_f - \mathbf{C}_{fp}\mathbf{C}_{pp}^{-1}\bar{\mathbf{x}}_p\right)$, where $^+$ indicates pseudo-inverse. The key idea of our first online algorithm is to keep track of $\mathbf{C}_{fp}\mathbf{C}_{pp}^{-1}$ via updating synaptic weights in the network. Figure 1 presents the details of this *circuit A* algorithm, where each node is a layer of neurons, and edges are synaptic connections. Each time step, we update the circuit through **steps 1-4** (left). The reciprocal lateral connections $\mathbf{M}_{fp}$

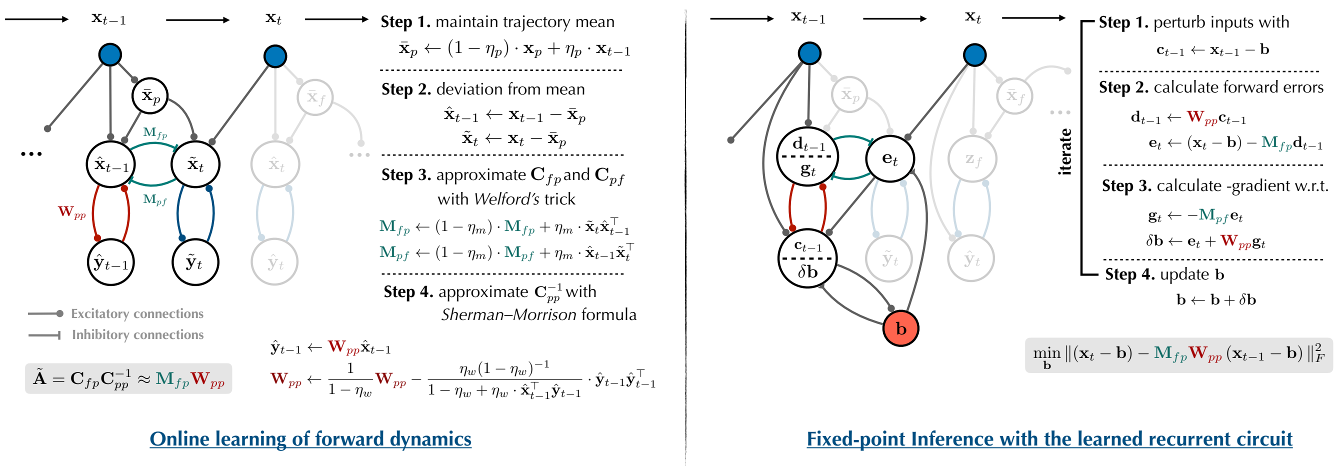

Figure 1: Neuronal *circuit A*: learning and inference algorithms.

and $\mathbf{M}_{pf}$ approximate covariance matrices with anti-Hebbian plasticity. Note that both pre- and post-synaptic neurons of lateral connections are subtracted by the past trajectory mean, which is Welford's trick [2] to keep unbiased variance estimation. The inverse of variance matrix $\mathbf{C}_{pp}^{-1}$ is maintained by connections $\mathbf{W}_{pp}$, which is updated by the Sherman-Morrison formula. Then, we reuse the learned recurrent circuit and iterate **steps 1-4** (right) to perform stochastic gradient descent to find the least-square estimate of $\mathbf{b}$. When the learning rates η_p, η_m, η_w decay as $1/t$ [3], with proper initialization, the *circuit A* returns the same inferred $\mathbf{b}$ at time T as the offline algorithm.

Neuronal *circuit B*. Our second online neural network algorithm is to directly solve $\tilde{\mathbf{A}}$ and $\mathbf{b}$ via gradient descent. As shown in Figure 2, $\tilde{\mathbf{A}}$ is maintained by the lateral connections and updated by anti-Hebbian plasticity, and $\mathbf{b}$ is maintained as the neural activity of the red neurons. Unlike *circuit A*, whose learning step 4 is not strictly local, the update rules of *circuit B* are all local.

$$\min_{\tilde{\mathbf{A}},\mathbf{b}} \frac{1}{T}\sum_{t=1}^{T} \|(\mathbf{x}_t - \mathbf{b}) - \tilde{\mathbf{A}}(\mathbf{x}_{t-1} - \mathbf{b})\|_2^2$$

$$\mathbf{y}_{t-1} \leftarrow \mathbf{x}_{t-1} - \mathbf{b}$$
$$\mathbf{e}_t \leftarrow \mathbf{x}_t - \mathbf{b} - \tilde{\mathbf{A}}\mathbf{y}_{t-1}$$
$$\mathbf{z}_t \leftarrow -\tilde{\mathbf{A}}^\top \mathbf{e}_t$$
$$\Delta\tilde{\mathbf{A}} \propto \sum_{t=1}^{T} \mathbf{e}_t \mathbf{y}_{t-1}^\top \qquad \Delta\mathbf{b} \propto \sum_{t=1}^{T} (\mathbf{e}_t + \mathbf{z}_t)$$

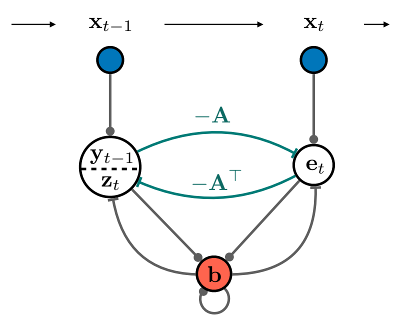

Figure 2: Neuronal *circuit B* update rules.

Experiment Results. We test our circuits on multiple classic dynamical systems, including three switching linear systems and two nonlinear systems (a four-fixed-points system; double pendulum). In Figure 3, trajectories are shown in dark blue, and the fixed points predicted by *circuit A* are shown in red. Stars indicate ground truth fixed points. For example, the double pendulum system has one stable fixed point (p_1) and three unstable fixed points (p_2, p_3, p_4) at angles $\theta_1, \theta_2 \in \{0, \pi\}$ with zero angular velocity. Even when systems are noisy, our *circuit A* can robustly predict the fixed points regardless of their types (stable/unstable/center/saddle) from every single trajectory. The heat map shows the distribution of the online fixed-point predictions. The *circuit B* converges much slower than *circuit A* (not shown).

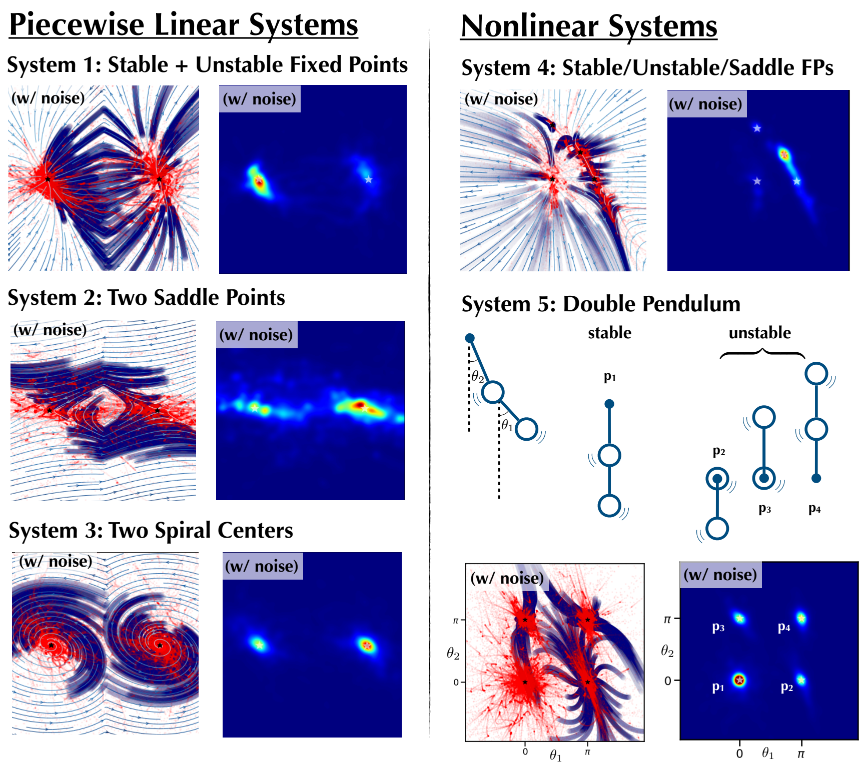

Figure 3: Trajectories (blue) and fixed-point estimates (red).

Conclusion. We derive two *biologically plausible* neural networks that can *efficiently* and *robustly* predict fixed point from a single trajectory. These networks are potential building blocks of a larger neuronal circuit for systems identification and model-based control, e.g., a bio-plausible online K-means [4] on top of them would learn all fixed points and transitions.

[1] Kutz et al. Dynamic mode decomposition: data-driven modeling of complex systems. SIAM, 2016. [2] Welford. Note on a method for calculating corrected sums of squares and products. Technometrics. 1962. [3] Robbins and Monro. Stochastic approximation method. The Annals of Mathematical Statistics. 1951. [4] Pehlevan and Chklovskii. A Hebbian/anti-Hebbian network derived from online non-negative matrix factorization can cluster and discover sparse features. IEEE, 2014.