

# Stage-like Emergence of Task Strategies in Animals and in Neural Networks Trained by Gradient Descent

J Tyler Boyd-Meredith, Cristofer Holobetz, and Andrew M Saxe  
Sainsbury Wellcome Centre, UCL

## Abstract

Humans and animals learning a task often appear to adopt a series of distinct strategies before reaching expert performance. This progression could result from deliberately testing distinct hypotheses about task contingencies. However, stage-like strategy changes can also be produced by artificial neural networks (ANN) learning by gradient descent (GD) without any explicit notion of task strategy. In this setting, apparent strategies correspond to saddle points in the loss dynamics around which learning slows before accelerating toward the next fixed point. We trained mice to perform a previously developed discrimination task, which they acquired in a series of stage-like behavioral transitions. We then developed an ANN model that recapitulated these transitions. By measuring the magnitude of the gradients during learning, we determined when the network approached (decreasing norm) and escaped saddle points (increasing norm) before reaching expert performance. Our modeling results show that even simple connectionist models without explicit hypotheses can be tailored to produce stages of learning that match what we observe in animals. We propose to develop and apply a method to identify saddle points of the loss and the likely transitions between them by performing gradient descent, not on the loss function, but on the magnitude of its gradient. For this abstract, we show how this tool identifies saddle points and their connections in a toy example.

**Keywords:** Learning Dynamics; Associative Learning; Hypothesis-testing

## Mice Progress Through Stereotyped Series of Strategies During Task Acquisition

We trained mice to perform a discrimination task in a virtual reality corridor developed by Sun et al. (2025). On each trial, 1 of 2 marked positions, called the ‘near’ and ‘far’ zones, was rewarded if the mouse licked a water spout while in the relevant zone (Fig. 1a). At the beginning of the trial, the mouse ran through an ‘indicator’ region with a pattern on the wall that differed depending on which zone had reward available.

Mice appeared to learn this task by progressing through a series of distinct strategies (Fig. 1a), as observed in (Sun et al., 2025). First, they licked the spout everywhere in the corridor, independent of their position (‘lick all’). Second, they licked reliably in both reward zones, regardless of trial type (‘lick both’). Third, they licked reliably in the ‘near’ zone, but withheld licking at the ‘far’ zone if they had already received reward (‘lick stop’). Finally, they learned an indicator-dependent

strategy where they only licked in the ‘near’ zone when instructed by the indicator (‘expert’).

## ANNs Learning by Gradient Descent Reproduce Stage-Like Strategy Acquisition

Inspired by recent work that interprets animal learning as governed by saddle point structure in a network’s loss landscape (Liebana Garcia et al., 2025), we sought to test whether we could reproduce this stage-like acquisition of strategies in a minimal artificial neural network (ANN) model of our task. To do this, we built a 2 layer network with ReLu activations in the hidden layer and a single sigmoidal output corresponding to a lick probability

$$P(\text{lick} | x) = \sigma(W_2(W_1x)^+) \quad (1)$$

where  $W_i$  indicates the  $i^{\text{th}}$  layer weights,  $(\cdot)^+$  indicates ReLu activation function,  $\sigma$  represents the logistic function, and  $x$  is the input (Fig 1b).

The strategies discovered by this model depended on how we featurized the inputs  $x$ . We discretized positions along the corridor and structured  $x$  as one input vector per position for each trial type with  $x_{\text{far}}^i$  representing the  $i^{\text{th}}$  position on a ‘far’ trial. To get the ‘lick all’ strategy, we needed a bias in  $x$  that was set to 1 regardless of trial type or position. To get the position- and reward-dependent ‘lick stop’ strategy, we added a one-hot encoding of the ‘near’ and ‘far’ positions, along with a memory for any reward delivered in the trial. Finally, to get indicator-dependent ‘expert’ behavior, we added a persistent one-hot encoding of which indicator was shown at the beginning of the trial.

The network’s objective function was the reward delivered for a given lick, minus the cost of licking

$$\mathcal{L}_{\text{near/far}}^i = -\left(ry_{\text{near/far}}^i - \kappa\right)\hat{y}_{\text{near/far}}^i \quad (2)$$

where  $r$  is the value of the reward,  $\kappa$  is the cost of licking,  $y_{\text{near/far}}^i \in \{0, 1\}$  indicates whether the  $i^{\text{th}}$  position on a given trial type is rewarded, and  $\hat{y}_{\text{near/far}}^i \in \{0, 1\}$  is a draw from a Bernoulli distribution with probability given by equation 1. After each lick, the weights were updated by gradient descent on equation 2.

ANN output activations throughout learning show a progression through the same strategies observed in the mouse data (Fig 1c). The average loss on each trial,  $\langle \mathcal{L} \rangle$ , showed plateaus corresponding to saddle point approach around the time of strategy transitions (Fig 1d, top). This is clearer in the magnitude of the gradients,

$$Q = \frac{1}{2} |\nabla_w \mathcal{L}|^2, \quad (3)$$

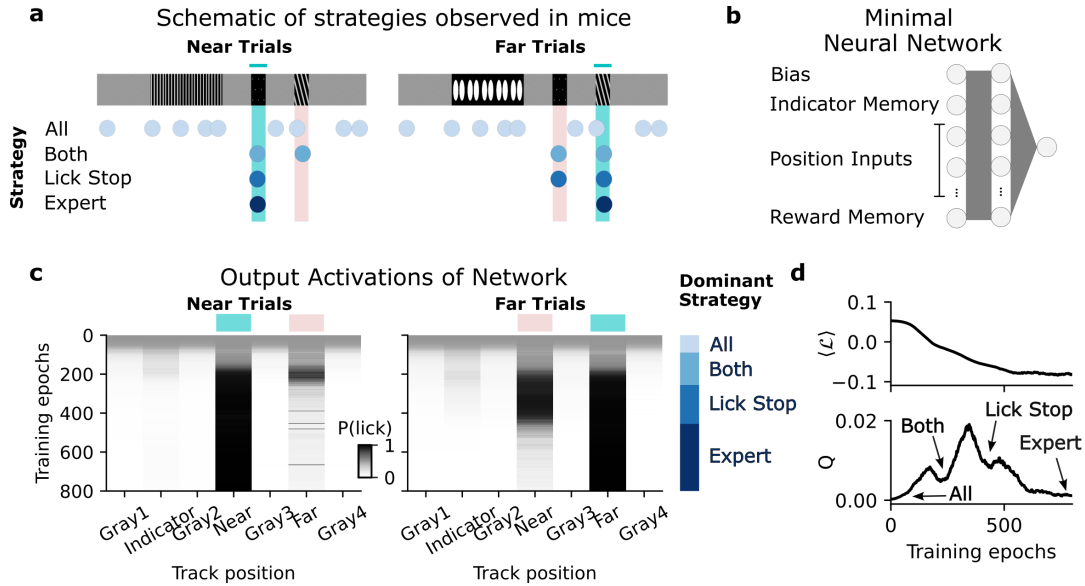


Figure 1: **ANN recapitulates stage-like strategy progression observed in mice** (a) Task schematic. (b) Network setup. (c) Output activations during learning. (d) Loss and gradient magnitude trajectories during learning.

which falls as the network approaches each fixed point and then rises during escape from saddle points (Fig 1d, bottom).

### Constructing Fixed Point Graphs with Flow Minimization and the String Method

We want to understand the dynamics of our model's behavior in terms of the fixed point structure of its loss landscape. One representation of this structure is a graph with nodes corresponding to fixed points and edges reflecting transitions between them (Liebana Garcia et al., 2025). To find fixed points, we use a technique from Sussillo and Barak (2013), which finds fixed points in dynamical systems by minimizing the magnitude of the flow in the system. We refer to this method as flow minimization (FM). In a network learning by gradient descent, this corresponds to minimizing equation 3. By minimizing the norm of the gradient instead of the loss, the optimizer is attracted to all regions of near-zero gradient, even saddles. Fixed points can then be classified using second order information. To identify likely transitions between fixed points, we next apply a technique called the string method (E, Ren, & Vanden-Eijnden, 2002), which finds minimum energy paths between pairs of points in weight space. We then complete the graph by adding edges between points whose minimum energy path monotonically decreases in loss.

**Toy Model: 2 Layer Linear Chain** To illustrate this technique, we consider the simplest deep neural network: two linear neurons in series, trained on the task  $y = x$ . This produces a saddle point at  $(0,0)$  and a hyperbolic solution manifold for all  $w_1, w_2$  where  $w_1 \cdot w_2 = 1$ . For any initialization, training this network by GD will discover a valid solution, but for initializations close to the saddle point, FM will find the saddle point. The string method identifies edges that correctly reflect the

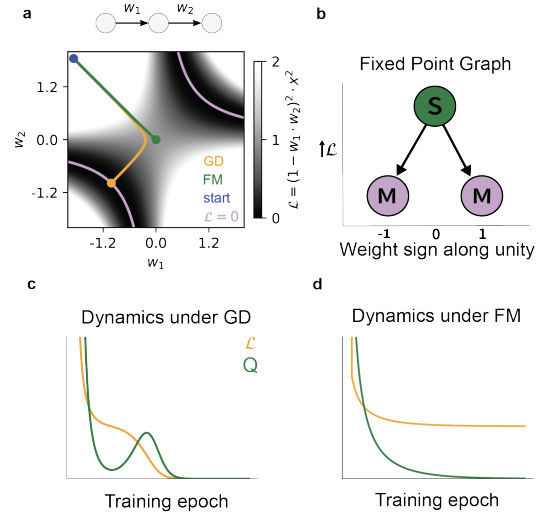


Figure 2: **Automatic Construction of a Fixed Point Graph** (a) Evolution of weights under GD and FM. (b) Relationships between fixed points. M and S refer to minima and saddles, respectively. Arrows denote directly connected pairs of fixed points. (c) Under GD,  $Q$  transiently drops as  $\mathcal{L}$  slows near the saddle point. (d) Under FM,  $Q$  approaches 0 monotonically, but  $\mathcal{L}$  plateaus as FM finds the saddle point.

bifurcation in weight trajectories across the two sides of the saddle point.

**Future Plans** We will apply this method to the ANN performing the discrimination task described above to produce a graph of its saddle point structure. This tool will be useful for predicting the stage-like strategy acquisition for a wide array of tasks and network architectures.

## References

- E, W., Ren, W., & Vanden-Eijnden, E. (2002, August). String method for the study of rare events. *Physical Review*. Retrieved from <https://doi.org/10.1103/PhysRevB.66.052301>  
doi: 10.1103/PhysRevB.66.052301
- Liebana Garcia, S., Laffere, A., Toschi, C., Schilling, L., Podlaski, J., Fritsche, M., ... Lak, A. (2025, June). Dopamine encodes deep network teaching signals for individual learning trajectories. *Cell*. Retrieved from <https://doi.org/10.1016/j.cell.2025.05.025> doi: 10.1016/j.cell.2025.05.025
- Sun, W., Winnubst, J., Natrajan, M., Lai, C., Kajikawa, K., Bast, A., ... Spruston, N. (2025, February). Learning produces an orthogonalized state machine in the hippocampus. *Nature*, *640*(8057), 165–175. Retrieved from <http://dx.doi.org/10.1038/s41586-024-08548-w>  
doi: 10.1038/s41586-024-08548-w
- Sussillo, D., & Barak, O. (2013, March). Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural Computation*, *25*(3), 626–649. Retrieved from [http://dx.doi.org/10.1162/NECO\\_a00409](http://dx.doi.org/10.1162/NECO_a00409) doi: 10.1162/neco\_a00409