A Simulation Procedures: Parameters and Test Settings

We used RELEARNN to train neural networks on two different tasks. The first task was to link visual contours and the second task was curve tracing. Parameters in both experiments have been set to \(\alpha = \beta = \zeta = 1\), \(\gamma = 4\), \(a = 0.001\) (c.f. eqn. (1)) and the reward was set to \(\varrho = 0\) and \(\varrho = 0.4\) in case of error and success, respectively. We performed simulations with the one step Euler-method with \(\Delta t = 0.04\) iterating until convergence \((\varepsilon = 0.0001)\). Weights were bounded by \(\theta^{\text{high}} = 4\) by adapting the weights according to \(W \mapsto W + \Delta W \cdot (\theta^{\text{high}} - W)\) to accommodate the fact that neural connectivity cannot be arbitrarily large. However, in practice the maximal value of any weight in the networks was \(\approx 2\). Excitatory and inhibitory connections to the linking layer were initialized with values randomly chosen from the interval \([0.98, 1.02]\). Connections to (excitatory) and from (modulatory) the association layer were initialized with small weights randomly chosen from the interval \([0.001, 0.02]\). To speed up learning in the tracing task each motor unit was “paired” with an association unit from which it received strong excitation randomly initialized with values uniformly drawn from \([0.95, 1.05]\). Training simulations conducted without this high initialization weights were also successful but took longer. The lateral modulatory connections were initialized with values randomly drawn from \([0.001, 0.05]\) (to speed up learning in the tracing task, we initialized them with \([1.45, 1.55]\)). The learning rate \(\eta\) was set to 0.65 and the parameter \(\tau\) of the softmax rule was set to 0.004.

A.1 Learning to Link Contours

We presented stimuli within two apertures with randomly oriented bars to the input layer of the network on a \(9 \times 9\) grid with units selective to four orientations. The activity \(r\) of a neuron tuned to orientation \(\Theta\) elicited by a bar with orientation \(\vartheta\) equalled \(r = \exp(\kappa \cdot \cos(2 \cdot (\vartheta - \Theta))) - \kappa - |\chi|\), with \(\kappa = .9\) and a Gaussian noise \(\chi\) with standard deviation 0.65 to account for the jitter used in the experiments of [14]. There were a total of two motor units that signaled the hemisfield with the target contour and four association layer units. We did not include interhemispheric connections between units in the linking layer that represented input patterns on either side of the fixation point.

A.2 Curve-Tracing Task

In the curve-tracing task, the visual input was presented on a \(5 \times 5\) grid of neurons selective for luminance and for the colors red and green. To decrease the computational costs, we did not model orientation selective responses as they do not encode information that was relevant for the present task. The linking layer also consisted of \(5 \times 5 \times 3\) neurons, receiving input from the units selective for luminance, red and green. The association layer contained 25 neurons and the motor layer consisted of a \(5 \times 5\) grid of possible saccade targets, because every pixel in the input layer could be selected for a saccade.

We started with a pre-training phase where we trained the model to make saccades to green dots, thereby emulating the shaping strategy used to train monkeys in comparable tasks. In the pre-training phase, we set the activity of the neurons selective for green and luminance to 0.7 and discontinued pre-training when a greedy decision policy (i.e. selection of the maximally active output neuron) would have succeeded for 400 times in a row.

In the curve-tracing task, we generated two lines of equal length. The activity of luminance input neurons was 0.1, the activity of the neurons coding red at the cued location was 0.7 and the activity
of neurons coding the green saccade targets was 0.7. We started with curves with a length of 2 and we gradually increased line length (choosing random configurations of adjacent pixels for the target and distractor curve) up to a length of 5 whenever a greedy-policy would have been successful for 400 stimulus presentations in a row.

B Taylor-Expansion

Here, we derive the Taylor expansion of the full dynamics given in eqn. (3). We define the right-hand side of eqn. (3) as

\[ f(p) = -\text{diag}(\alpha)p + \text{diag}(\beta - p) \cdot \text{diag}((W^{ex})^T g(p) + (W^I)^T I^{inp}) \cdot (1 + \gamma (W^{mod})^T g(p)) - \text{diag}(\zeta + p) \cdot (W^{inh})^T g(p). \] (35)

This is, for a single element \( p_i \)

\[ f(p_i) = -\alpha p_i + (\beta - p_i) \cdot \left( \sum_{j=1}^{N} W_{ji}^{ex} g(p_j) + \sum_{j=1}^{N} W_{j}^{I} I_{j}^{inp} \right) \cdot \left( 1 + \gamma \cdot \sum_{j=1}^{N} W_{ji}^{mod} g(p_j) \right) - \]

\[ (\zeta + p_i) \cdot \sum_{j=1}^{N} W_{ji}^{inh} g(p_j). \] (36)

To determine the Taylor expansion of \( f \) up to first order, we only need to determine \( d/dp f \) since \( f(p^\infty) = 0 \). Derivation with respect to \( p \) according to matrix calculus results in

\[ \frac{d}{dp} f(p) = -\text{diag}(\alpha) - \text{diag} (I^{ex} \cdot (1 + \gamma I^{mod}) + \text{diag} ((\beta - p)(1 + \gamma I^{mod})) (W^{ex})^T G - \gamma \cdot \text{diag} ((\beta - p)I^{ex})(W^{mod})^T G - \text{diag}(I^{inh}) - \text{diag}(\zeta + p)(W^{inh})^T G =: \mathcal{L}^T, \] (37)

with

\[ G = \text{diag} \left[ \frac{d}{dp} g(p) \right]. \] (38)

C Connection to Almeida-Pineda: Derivation of Supervised Learning in Recurrent Neural Networks (LEARNN)

In this section, we apply the Almeida-Pineda algorithm [82,87] for the neuron type described in the Models section. We derive a supervised learning algorithm for recurrent neural networks which we call LEARNN. We will then adapt the teaching signal to learn from experience and demonstrate that LEARNN in this reinforcement setting is identical to the RELEARNN algorithm using gated Hebbian plasticity.

As in the Almeida-Pineda method, we assume that the system eqn. (3) reaches a stable steady state \( p^\infty \). Although little is known about the stability of the network system eqn. (3) for arbitrary weight matrices our numerical simulations confirmed the observations of [82] and [109] of similar systems that
the dynamics tend to converge to a stable state. Furthermore, the boundedness of the activities in system eqn. (3) guarantees that the system stays within these bounds. The learning mechanism minimizes an error function that reflects deviation from the target values $\tau_j$

$$E(x) = \frac{1}{2} \sum_{i=1}^{N} J_i^2, \quad \text{where} \quad J_i = \begin{cases} \tau_i - p_i^\infty, & i \in \mathcal{O}, \\ 0, & \text{otherwise.} \end{cases} \quad (39)$$

Weight changes are calculated by gradient-descent with learning rate $\eta$ ($\eta$ must be small so that $p$ is essentially at steady state, i.e. $p \approx p^\infty$)

$$\Delta W_{kl} = -\eta \frac{\partial E}{\partial W_{kl}^\infty} = \eta \sum_{i=1}^{N} J_i \frac{\partial p_i^\infty}{\partial W_{kl}^\infty}. \quad (40)$$

In the next step, we derive the values of $\frac{\partial p_i^\infty}{\partial W_{kl}^\infty}$, i.e. how a small change of weight $W_{kl}$ influences the activity of neuron $p_i$ at equilibrium. Applying matrix calculus (e.g. [89]; c.f. appendices C.1 to C.4), we calculate the weight changes such that

$$\frac{\partial p_i^\infty}{\partial W_{kl}^\text{imp}} = -(\mathcal{L}^{-1})_{ii} \cdot (\beta - p_i^\infty) \cdot (1 + \gamma (I_{\infty}^{\text{imp}})_i) \cdot p_{k}^\text{imp}, \quad (41)$$

$$\frac{\partial p_i^\infty}{\partial W_{kl}^\text{ex}} = -(\mathcal{L}^{-1})_{ii} \cdot (\beta - p_i^\infty) \cdot (1 + \gamma (I_{\infty}^{\text{ex}})_i) \cdot g(p_k^\infty), \quad (42)$$

$$\frac{\partial p_i^\infty}{\partial W_{kl}^\text{inh}} = (\mathcal{L}^{-1})_{ii} \cdot (\zeta + p_i^\infty) \cdot g(p_k^\infty), \quad (43)$$

$$\frac{\partial p_i^\infty}{\partial W_{kl}^\text{mod}} = -\gamma \cdot (\mathcal{L}^{-1})_{ii} \cdot (\beta - p_i^\infty) \cdot (I_{\infty}^{\text{ex}})_i \cdot g(p_k^\infty). \quad (44)$$

where\(^1\)

$$\mathcal{L}^T = -\text{diag}[\alpha] - \text{diag} \left[I_{\infty}^\text{ex} \cdot \left(1 + \gamma I_{\infty}^{\text{mod}}\right)\right] - \text{diag} \left[I_{\infty}^{\text{inh}}\right] + \text{diag} \left[\left(\beta - p^\infty\right) \cdot \left(1 + \gamma I_{\infty}^{\text{mod}}\right)\right] (W^\text{ex})^T G - \text{diag} \left[\zeta + p^\infty\right] \cdot (W^{\text{inh}})^T G + \gamma \cdot \text{diag} \left[\left(\beta - p^\infty\right) \cdot I_{\infty}^{\text{ex}}\right] \cdot (W^{\text{mod}})^T G \quad (45)$$

with

$$G = \text{diag} \left[\frac{d}{dp} g(p)\right]. \quad (46)$$

See sections C.1 to C.4 for a detailed derivation of eqns. (41) to (45). Here, $\text{diag}[\alpha] \in \mathbb{R}^{N \times N}$, $\alpha \in \mathbb{R}$ denotes a diagonal matrix, where each element $a_{ii}$ has the value $\alpha$ and $\text{diag}[p] \in \mathbb{R}^{N \times N}$, $p \in \mathbb{R}^N$ is a diagonal matrix with $a_{ii} = p_i$. Note that eqns. (41) to (44) require a matrix inversion. Instead of calculating the matrix inverse, we introduce an associated dynamical system utilizing a population of neurons $n_j^z, j = 1 \ldots N$ with activities $z_j$

$$\dot{z} = \mathcal{L} z + J \quad (47)$$

\(^1\)Note that this $L$ is identical to the $L$ used in the derivation of the gated Hebbian plasticity but that it is derived by calculating the derivative of $p^\infty$ with respect to the weights.
where $J$ denotes the deviation of activations from the target values eqn. (39) and $L$ is defined as in eqn. (45). At steady state the following algebraic condition holds:

$$z_i^\infty = -(L^{-1}J)_i = -\sum_{i=1}^{N}(L^{-1})_{ii}J_i.$$  \hfill (48)

This steady state equation can be utilized to avoid computing the matrix inverse in eqns. (41) to (44). The resulting learning rules can now be derived as local mechanisms by combining eqns. (41) to (44), eqn. (40) and eqn. (48):

$$\Delta W_{kl}^{\text{Inp}} = \eta \cdot z_i^\infty \cdot (\beta - p_i^\infty) \cdot (1 + \gamma(I_{\text{ex}}^\infty)_i) \cdot (1 + \gamma(I_{\text{mod}}^\infty)_i) \cdot p_{kl}^{\text{inp}},$$  \hfill (49)

$$\Delta W_{kl}^{\text{ex}} = \eta \cdot z_i^\infty \cdot (\beta - p_i^\infty) \cdot (1 + \gamma(I_{\text{mod}}^\infty)_i) \cdot g(p_k^\infty),$$  \hfill (50)

$$\Delta W_{kl}^{\text{inh}} = -\eta \cdot z_i^\infty \cdot (\zeta + p_i^\infty) \cdot g(p_k^\infty),$$  \hfill (51)

$$\Delta W_{kl}^{\text{mod}} = \eta \cdot \gamma \cdot z_i^\infty \cdot (\beta - p_i^\infty) \cdot (I_{\text{ex}}^\infty)_i \cdot g(p_k^\infty),$$  \hfill (52)

where $z_i^\infty$ is the steady state eqn. (48) of the associated network denoted in eqn. (47).

Equations (3), (47) and (49) to (52) completely specify the dynamics of LEARNNN provided that (3) and (47) are convergent. It is known that convergence of a system of neurons with activation dynamics as defined in eqn. (3) is a sufficient condition for the convergence of (47) [87,88]. To ensure boundedness, positivity of the elements of the weight matrices must be enforced in each step, i.e. weights are rectified according to $W_{ij} = \max\{0, W_{ij}\}$.

Now LEARNNN can be easily extended to reinforcement learning. Like described in the Models section, the network acts as a reinforcement learning agent. The state is presented as input to the network and the output units engage in a competition that selects one of a number of actions. We employ a SARSA-style return as in eqn. (7) and define the teaching signal of LEARNNN as

$$J_i = \begin{cases} 
\rho - p_a(s), & i = a, \\
0, & i \neq a.
\end{cases}$$  \hfill (53)

We can now compare eqn. (50) (LEARNNN; recurrent gradient descent) and eqn. (8). When we factor out the SARSA-term for immediate rewards $\delta := \rho - p_a(s)$ in eqn. (48), we note that LEARNNN in the reinforcement setting (recurrent gradient descent) is identical to RELEARNNN (gated Hebbian plasticity).

### C.1 Derivative with Respect to the Input Weights

This section demonstrates how eqn. (41) is derived. We start with the derivative $\frac{\partial}{\partial W_{kl}^{\text{Inp}}} p^\infty$: At equilibrium, $\frac{d}{dt}p = 0$, so that eqn. (3) changes to

$$0 = -\text{diag}(\alpha)p^\infty + \text{diag}(\beta - p^\infty) \cdot \text{diag}((W_{\text{ex}}^T g(p^\infty) + (W_{\text{mod}}^T I_{\text{ex}}^{\text{inp}}) \cdot (1 + \gamma(W_{\text{mod}}^T) g(p^\infty)) - \text{diag}(\zeta + p^\infty) \cdot (W_{\text{inh}}^T g(p^\infty)) \cdot I_{\text{mod}}^{\text{inh}} .$$  \hfill (54)
For simplicity, we omit the $\infty$ and just write $p, I^{ex}$ instead of $p^{\infty}, I^{ex}_{\infty}$. Derivation with respect to $W_{kl}^{inp}$ according to matrix calculus results in

$$0 = -\text{diag}(\alpha) \frac{\partial}{\partial W_{kl}^{inp}} p - \text{diag}(I^{ex}) \text{diag} \left( 1 + \gamma I^{mod} \right) \frac{\partial}{\partial W_{kl}^{inp}} p$$

$$+ \text{diag}(\beta - p) \text{diag} \left( 1 + \gamma I^{mod} \right) \left( (W^{ex})^T G \frac{\partial}{\partial W_{kl}^{inp}} p + E_{lk} p_{inp} \right)$$

$$+ \gamma \cdot \text{diag}(\beta - p) \text{diag}(I^{ex}) (W^{mod})^T G \frac{\partial}{\partial W_{kl}^{inp}} p$$

$$- \text{diag}(I^{inh}) \frac{\partial}{\partial W_{kl}^{inp}} p - \text{diag}(\zeta + p) (W^{inh})^T G \frac{\partial}{\partial W_{kl}^{inp}} p$$

(55)

where $E_{lk} \in \mathbb{R}^{N \times N}$ is a matrix with zeros except for $(E_{lk})_{lk} = 1$, $e_{l} \in \mathbb{R}^{N}$ is a vector with zeros except for $(e_{l})_{l} = 1$ and $G$ as in eqn. (46). We can now gather all factors of $\frac{\partial}{\partial W_{kl}^{inp}} p$ and take them to the left-side.

$$\left\{ \text{diag}(\alpha) + \text{diag} \left[ I^{ex} \cdot (1 + \gamma I^{mod}) \right] - \text{diag} \left[ (\beta - p)(1 + \gamma I^{mod}) \right] (W^{ex})^T G - \right.$$  

$$\gamma \cdot \text{diag} \left[ (\beta - p) \cdot I^{ex} \right] (W^{mod})^T G + \right.$$  

$$\text{diag}(I^{inh}) + \text{diag}(\zeta + p) (W^{inh})^T G \right\} \frac{\partial}{\partial W_{kl}^{inp}} p =$$  

$$\text{diag}(\beta - p) \text{diag} \left( 1 + \gamma I^{mod} \right) e_{l} \cdot p_{k}^{inp}.$$  

(56)

When we define the term in curled brackets as $-L^T$, we obtain eqn. (45). Multiplication with $-L^{-1}$ results in eqn. (41).

### C.2 Derivative with Respect to the Excitatory Weights

This section demonstrates how eqn. (42) is derived. We start with the derivative $\frac{\partial}{\partial W_{kl}^{ex}} p^{\infty}$: At equilibrium, $\frac{dp}{dt} = 0$, so that eqn. (3) changes to eqn. (54) (repeated for reader’s convenience):

$$0 = -\text{diag}(\alpha) p^{\infty} +$$

$$\text{diag}(\beta - p^{\infty}) \cdot \text{diag} \left( (W^{ex})^T g(p^{\infty}) + (W^{mod})^T I^{inp} \right) \cdot (1 + \gamma \left( W^{mod} \right)^T g(p^{\infty}) -$$

$$\text{diag}(\zeta + p^{\infty}) \cdot (W^{inh})^T g(p^{\infty}) \right). (57)$$
For simplicity, we omit the $\infty$ and just write $p, I^{ex}$ instead of $p^\infty, I^{ex}_\infty$. Derivation with respect to $W_{kl}^{ex}$ according to matrix calculus results in

$$0 = -\text{diag}(\alpha) \frac{\partial}{\partial W_{kl}^{ex}} p - \text{diag}(I^{ex}) \text{diag} \left( 1 + \gamma I^{mod} \right) \frac{\partial}{\partial W_{kl}^{ex}} p$$

$$+ \text{diag}(\beta - p) \text{diag} \left( 1 + \gamma I^{mod} \right) \left( \frac{E_{ikg}(p) + (W^{ex})^T G - \partial}{\partial W_{kl}^{ex}} p \right)$$

$$+ \gamma \cdot \text{diag}(\beta - p) \text{diag}(I^{ex})(W^{mod})^T G \frac{\partial}{\partial W_{kl}^{ex}} p$$

$$- \text{diag}(W^{inh}) \frac{\partial}{\partial W_{kl}^{ex}} p - \text{diag}(\zeta + p)(W^{inh})^T G \frac{\partial}{\partial W_{kl}^{ex}} p,$$

(57)

where $E_{ik} \in \mathbb{R}^{N \times N}$ is a matrix with zeros except for $(E_{ik})_{ik} = 1$, $e_l \in \mathbb{R}^N$ is a vector with zeros except for $(e_l)_l = 1$ and $G$ as in eqn. (46). We can now gather all factors of $\frac{\partial}{\partial W_{kl}^{ex}} p$ and take them to the left-side.

$$\left\{ \text{diag}(\alpha) + \text{diag} \left[ I^{ex} \cdot (1 + \gamma I^{mod}) \right] - \text{diag} \left[ (\beta - p)(1 + \gamma I^{mod}) \right] (W^{ex})^T G - \right.$$

$$\gamma \cdot \text{diag} \left[ (\beta - p) \cdot I^{ex} \right] (W^{mod})^T G +$$

$$\text{diag}(I^{inh}) + \text{diag}(\zeta + p)(W^{inh})^T G \right\} \frac{\partial}{\partial W_{kl}^{ex}} p =$$

$$\text{diag}(\beta - p) \text{diag} \left( 1 + \gamma I^{mod} \right) e_l \cdot g(p_k) .$$

(58)

When we define the term in curled brackets as $-\mathcal{L}^T$, we obtain eqn. (45). Multiplication with $-\mathcal{L}^{-1}$ results in eqn. (42).

### C.3 Derivative with Respect to the Inhibitory Weights

This section demonstrates how eqn. (43) is derived. We start with the derivative $\frac{\partial}{\partial W_{kl}^{inh}} p^\infty$: At equilibrium, $\frac{d}{dp} p = 0$, so that eqn. (3) changes to eqn. (54) (repeated for reader’s convenience):

$$0 = -\text{diag}(\alpha) p^\infty +$$

$$\text{diag}(\beta - p^\infty) \cdot \text{diag} \left( \frac{(W^{ex})^T g(p^\infty) + (W^{inp})^T I^{inp}}{I^{ex}_\infty} \right) (1 + \gamma \frac{(W^{mod})^T g(p^\infty)}{I^{mod}_\infty}) -$$

$$\text{diag}(\zeta + p^\infty) \cdot \frac{(W^{inh})^T g(p^\infty)}{I^{inh}_\infty} .$$
For simplicity, we omit the $\infty$ and just write $p, I^e$ instead of $p^\infty, I^e_\infty$. Derivation with respect to $W^\text{inh}_{kl}$ according to matrix calculus results in

$$
0 = -\operatorname{diag}(\alpha) \frac{\partial}{\partial W^\text{inh}_{kl}} p - \operatorname{diag}(I^e) \operatorname{diag} \left( 1 + \gamma I^\text{mod} \right) \frac{\partial}{\partial W^\text{inh}_{kl}} p \\
+ \operatorname{diag}(\beta - p) \operatorname{diag} \left( 1 + \gamma I^\text{mod} \right) (W^e)^T G \frac{\partial}{\partial W^\text{inh}_{kl}} p \\
+ \gamma \cdot \operatorname{diag}(\beta - p) (W^\text{mod})^T G \frac{\partial}{\partial W^\text{inh}_{kl}} p \\
- \operatorname{diag}(I^\text{inh}) \frac{\partial}{\partial W^\text{inh}_{kl}} p - \operatorname{diag}(\zeta + p) \left( (W^\text{inh})^T G \frac{\partial}{\partial W^\text{inh}_{kl}} p + \sum_{E_{lk}} e_{lg}(r) \right) ,
$$

(59)

where $E_{lk} \in \mathbb{R}^{N \times N}$ is a matrix with zeros except for $(E_{lk})_{lk} = 1$, $e_l \in \mathbb{R}^N$ is a vector with zeros except for $(e_l)_l = 1$ and $G$ as in eqn. (46). We can now gather all factors of $\frac{\partial}{\partial W^\text{inh}_{kl}} p$ and take them to the left-side.

$$
\left\{ \operatorname{diag}(\alpha) + \operatorname{diag} \left[ I^e \cdot (1 + \gamma I^\text{mod}) \right] - \operatorname{diag} \left[ (\beta - p)(1 + \gamma I^\text{mod}) \right] (W^e)^T G - \gamma \cdot \operatorname{diag} \left[ (\beta - p) \cdot I^e \right] (W^\text{mod})^T G + \operatorname{diag}(I^\text{inh}) + \operatorname{diag}(\zeta + p) (W^\text{inh})^T G \right\} \frac{\partial}{\partial W^\text{inh}_{kl}} p = -\operatorname{diag}(\zeta + p) e_l \cdot g(p_k).
$$

(60)

When we define the term in curled brackets as $-L^T$, we obtain eqn. (45). Multiplication with $-L^{T-1}$ results in eqn. (43).

### C.4 Derivative with Respect to the Modulatory Weights

This section demonstrates how eqn. (44) is derived. We start with the derivative $\frac{\partial}{\partial W^\text{mod}_{kl}} p^\infty$: At equilibrium, $\frac{d}{dt} p = 0$, so that eqn. (3) changes to eqn. (54) (repeated for reader’s convenience):

$$
0 = -\operatorname{diag}(\alpha) p^\infty + \operatorname{diag}(\beta - p^\infty) \cdot \operatorname{diag} \left( (W^e)^T g(p^\infty) + (W^i)^T I^\text{inp} \right) \cdot \underbrace{(1 + \gamma (W^\text{mod})^T g(p^\infty))}_{I^\text{mod}_\infty} \\
- \operatorname{diag}(\zeta + p^\infty) \cdot \underbrace{(W^\text{inh})^T g(p^\infty)}_{I^\text{inh}_\infty}.
$$
For simplicity, we omit the $\infty$ and just write $p, I^\text{ex}$ instead of $p^\infty, I^\text{ex}_\infty$. Derivation with respect to $W_{kl}^\text{mod}$ according to matrix calculus results in

\[ 0 = -\text{diag}(\alpha) \frac{\partial}{\partial W_{kl}^\text{mod}} p - \text{diag}(I^\text{ex}) \frac{\partial}{\partial W_{kl}^\text{mod}} p \]

\[ + \text{diag}(\beta - p) \frac{\partial}{\partial W_{kl}^\text{mod}} (1 + \gamma I^\text{mod}) (W^\text{ex})^T G \frac{\partial}{\partial W_{kl}^\text{mod}} p \]

\[ + \gamma \cdot \text{diag}(\beta - p) \text{diag}(I^\text{ex}) \left( \frac{E_{lk} g(p)}{=e_l g(p_k)} + (W^\text{mod})^T G - \frac{\partial}{\partial W_{kl}^\text{mod}} p \right) \]

\[ - \text{diag}(I^\text{inh}) \frac{\partial}{\partial W_{kl}^\text{mod}} p - \text{diag}(\zeta + p) (W^\text{inh})^T G - \frac{\partial}{\partial W_{kl}^\text{mod}} p, \quad (61) \]

where $E_{lk} \in \mathbb{R}^{N \times N}$ is a matrix with zeros except for $(E_{lk})_{lk} = 1$, $e_l \in \mathbb{R}^N$ is a vector with zeros except for $(e_l)_l = 1$ and $G$ as in eqn. (46). We can now gather all factors of $\frac{\partial}{\partial W_{kl}^\text{mod}} p$ and take them to the left-side.

\[ \left\{ \text{diag}(\alpha) + \text{diag}(I^\text{ex} \cdot (1 + \gamma I^\text{mod})) - \text{diag}((\beta - p)(1 + \gamma I^\text{mod})) (W^\text{ex})^T G - \gamma \cdot \text{diag}((\beta - p) \cdot I^\text{ex}) (W^\text{mod})^T G + \right. \]

\[ \left. \text{diag}(I^\text{inh}) + \text{diag}(\zeta + p) (W^\text{inh})^T G \right\} \frac{\partial}{\partial W_{kl}^\text{mod}} p = \]

\[ \gamma \cdot \text{diag}(\beta - p) \text{diag}(I^\text{ex}) e_l \cdot g(p_k). \quad (62) \]

When we define the term in curled brackets as $-\mathcal{L}^T$, we obtain eqn. (45). Multiplication with $-\mathcal{L}^{T-1}$ results in eqn. (44).