

1 Writing out the whole system

The network dynamics are described as follows:

$$\begin{aligned}\frac{d\theta_i}{dt} &= (1 - \cos \theta_i) + (1 + \cos \theta_i) \cdot (\eta_i + I_i) \quad \theta_i \in \mathbb{R}^N \\ I_i &= \frac{\kappa}{\langle k \rangle} \sum_{j=1}^N A_{ij} P_n(\theta_j) \\ P_n(\theta_j) &= (1 - \cos \theta_j)\end{aligned}\tag{1}$$

We observe synchronization through the order parameter

$$Z(t) = \frac{1}{N} \sum_{j=1}^N e^{i\theta_j} \quad Z(t) \in \mathbb{C}\tag{2}$$

For a fixed degree network it has been proven that the order parameter follows:

$$\dot{Z}(t) = -i \frac{(Z - 1)^2}{2} + \frac{(Z + 1)^2}{2} \cdot \left(-\Delta + i\eta_0 + i\kappa \cdot \left(1 + \frac{Z^2 + \bar{Z}^2}{6} - \frac{4}{3} \text{Re}(Z) \right) \right)\tag{3}$$

For an arbitrary network the order parameter follows a trajectory per degree. When we assemble the whole expression for the Ott-Antonsen manifold as found in [?] with $H_2(\mathbf{k}, t)$ as in [?], we obtain the following:

$$\begin{aligned}\frac{\partial z(\mathbf{k}, t)}{\partial t} &= -i \frac{(z(\mathbf{k}, t) - 1)^2}{2} + \frac{(z(\mathbf{k}, t) + 1)^2}{2} \cdot I(\mathbf{k}) \quad z(\mathbf{k}, t) \in \mathbb{C}^{M_k} \\ I(\mathbf{k}) &= -\Delta(\mathbf{k}) + i\eta_0(\mathbf{k}) + i\kappa \cdot H_2(\mathbf{k}, t) \\ H_2(\mathbf{k}, t) &= \frac{1}{\langle k \rangle} \sum_{\mathbf{k}'} P(\mathbf{k}') a(\mathbf{k}' \rightarrow \mathbf{k}) \cdot \left(1 + \frac{z(\mathbf{k}', t)^2 + \bar{z}(\mathbf{k}', t)^2}{6} - \frac{4}{3} \text{Re}(z(\mathbf{k}', t)) \right)\end{aligned}\tag{4}$$

\mathbf{k} represents a two-dimensional vector of the in- an out degree as $\mathbf{k} = (k^{\text{in}}, k^{\text{out}})$ and has unique entries as it forms the support of the vector of degrees of (??) as $\deg(\theta_i)$. So $z(\mathbf{k}, t)$ is really a vector in the complex plane that represents the mean-field dynamics on any node with degree \mathbf{k} , so we could also index as $z(t)_{\mathbf{k}}$. \mathbf{k}' represents \mathbf{k} when \mathbf{k} is already in use.

We can then find the mean field dynamics through

$$\bar{Z}(t) = \frac{1}{N} \sum_{\mathbf{k}} P(\mathbf{k}) z(\mathbf{k}, t) \quad \bar{Z}(t) \in \mathbb{C}\tag{5}$$

It is important to notice that in (??) and (??) we actually compute an inner vector product, which is non-commutative for complex numbers:

$$a \cdot b = \overline{b \cdot a} \quad a, b \in \mathbb{C}^r\tag{6}$$

This is the result of the *Conjugate* or *Hermitian* symmetry of the inner product. This is especially important in the MATLAB implementation.

2 Initial conditions

As the systems in ????????? describe the same dynamics for fully connected networks, it is important to be able to transform initial conditions between systems. When transforming from $\theta_i(t) \rightarrow z(\mathbf{k}, t) \rightarrow Z(t)$ we go from $\mathbb{R}^N \rightarrow \mathbb{C}^{M_k}$ to $\mathbb{C}^{M_k} \rightarrow \mathbb{C}$. If we have the same initial conditions, then all systems will predict the same behaviour. We will only map everything to \mathbb{C} .

The following maps can be used to transform the initial conditions, but as they do not give any qualitative information on the dynamics or distributions of the variables, they are not valid for transforming between dynamics. We discard $t = 0$ for clarity.

Mapping operations onto the order parameter in the complex plane is straightforward:

$$\begin{aligned}\theta_i &\longrightarrow Z = \frac{1}{N} \sum_{j=1}^N e^{i\theta_j} \\ z(\mathbf{k}) &\longrightarrow Z = \frac{1}{N} \sum_{\mathbf{k}} P(\mathbf{k}) z(\mathbf{k}, t)\end{aligned}$$

Taking the inverse maps, we can make use of the fact that the average of a set of identical values is the value itself. For $z(\mathbf{k})$ we have a weighed average which we need to undo, making sure that the whole sums up to N .

$$\begin{aligned}Z &\longrightarrow \theta_i = -i \cdot \log(Z) \\ Z &\longrightarrow z(\mathbf{k}) = \frac{\overline{Z \cdot n(\mathbf{k})}}{P(\mathbf{k})}\end{aligned}$$

Then, transferring between θ_i and $z(\mathbf{k})$, we need to filter θ_i per degree:

$$\begin{aligned}z(\mathbf{k}) &\longrightarrow \theta_i = -i \cdot \log\left(\frac{z(\mathbf{k}) \cdot P(\mathbf{k})}{n(\mathbf{k})}\right) \quad \text{if } \deg(\theta_i) = \mathbf{k} \\ \theta_i &\longrightarrow z(\mathbf{k}) = \sum_{\mathbf{k}} e^{i\vartheta_{\mathbf{k}}} \quad \vartheta_{\mathbf{k}} = \sum_{i=1}^{n(\mathbf{k})} \theta_i \in \{\theta_i | \deg(\theta_i) = \mathbf{k}, \forall i \leq N\}\end{aligned}$$

Or also $\sum_i^N A_{ij} = k^{\text{in}} \cup \sum_j^N A_{ij} = k^{\text{out}}$. We can see how $\lim_{N \rightarrow +\infty} n(\mathbf{k}) = P(\mathbf{k})$, which makes these maps exact for any network size.

3 Fixpoint iteration

In [?] a fixpoint iteration is suggested to find attractive fixpoints of the system (??). If we set $\frac{\partial z(\mathbf{k}, t)}{\partial t} = 0$ we can solve the following system:

$$\begin{aligned}i \frac{(z(\mathbf{k}, t) - 1)^2}{2} &= \frac{(z(\mathbf{k}, t) + 1)^2}{2} \cdot I(\mathbf{k}) \\ i \left(\frac{z(\mathbf{k}, t) - 1}{z(\mathbf{k}, t) + 1} \right)^2 &= I(\mathbf{k}) \\ \frac{z(\mathbf{k}, t) - 1}{z(\mathbf{k}, t) + 1} &\equiv b(\mathbf{k}, t) \\ z(\mathbf{k}, t) - 1 &= b(\mathbf{k}, t) z(\mathbf{k}, t) + b(\mathbf{k}, t) \\ z(\mathbf{k}, t) \cdot (1 - b(\mathbf{k}, t)) &= b(\mathbf{k}, t) + 1\end{aligned}$$

We can then obtain the stable equilibria from:

$$ib(\mathbf{k}, t)^2 = I(\mathbf{k}) \quad z(\mathbf{k}, t)_{\pm} = \frac{1 \pm b(\mathbf{k}, t)}{1 \mp b(\mathbf{k}, t)} \quad (7)$$

where the signs are chosen so that $|z(\mathbf{k}, t)| \leq 1$.

4 A Newton-Raphson iteration for all fixpoints

4.1 Theory behind the method

The fixpoint iteration only gives us the stable equilibria of the system (??). We can obtain all equilibria and the Jacobian from a Newton-Raphson iteration. We define the equilibria $\mathbf{x}^* \in \mathbb{R}^n$ of a multivariate function $\mathbf{f}(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ with $\mathbf{f}(\mathbf{x}) = \mathbf{0}$. Expanding \mathbf{f} as a Taylor series, we obtain:

$$f_i(\mathbf{x} + \delta\mathbf{x}) = f_i(\mathbf{x}) + \sum_{j=1}^n \frac{\partial f_i(\mathbf{x})}{\partial x_j} \delta x_j + O(\delta\mathbf{x}^2) \approx f_i(\mathbf{x}) + \sum_{j=1}^n \frac{\partial f_i(\mathbf{x})}{\partial x_j} \delta x_j, \quad (i = 1, \dots, n) \quad (8)$$

We can also write this in vector notation, by setting $\mathbf{J}(\mathbf{x}) = \nabla \mathbf{f}(\mathbf{x}) = \frac{d}{d\mathbf{x}} \mathbf{f}(\mathbf{x}) \in \mathbb{R}^{n \times n}$

$$\mathbf{f}(\mathbf{x} + \delta\mathbf{x}) \approx \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_N(\mathbf{x}) \end{bmatrix} + \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_N}{\partial x_1} & \cdots & \frac{\partial f_N}{\partial x_N} \end{bmatrix} \begin{bmatrix} \delta x_1 \\ \vdots \\ \delta x_N \end{bmatrix} = \mathbf{f}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\delta\mathbf{x} \quad (9)$$

By assuming $\mathbf{f}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{0}$ we can find that $\delta\mathbf{x} = -\mathbf{J}^{-1}(\mathbf{x})\mathbf{f}(\mathbf{x})$ so that $\mathbf{x} + \delta\mathbf{x} = \mathbf{x} - \mathbf{J}^{-1}(\mathbf{x})\mathbf{f}(\mathbf{x})$. This expression converges to \mathbf{x}^* . When the equations are nonlinear, the equations converge to the real root as $\mathbf{x}_k = \mathbf{x}_k - \mathbf{J}^{-1}(\mathbf{x}_k)\mathbf{f}(\mathbf{x}_k)$.

For (??), we can compute the Jacobian for the diagonal and off-diagonal elements separately. But as $z(\mathbf{k}, t)$ is a complex function, first we need to understand what the derivative of a complex function is.

4.2 Derivatives of complex functions

For $z = x + iy \in \mathbb{C}$ and $x, y \in \mathbb{R}$ the conjugate is defined as $\bar{z} = x - iy$. That means that we can write the real and imaginary parts as:

$$x = \frac{z + \bar{z}}{2} \text{ and } y = -i \frac{z - \bar{z}}{2}$$

Using the chain rule, we can write the partial derivative with respect to z in function of x and y as x and y are functionally independent and find the first Wirtinger operator:

$$\frac{\partial}{\partial z} = \frac{\partial x}{\partial z} \frac{\partial}{\partial x} + \frac{\partial y}{\partial z} \frac{\partial}{\partial y} \longrightarrow \frac{\partial x}{\partial z} = \frac{1}{2} \text{ and } \frac{\partial y}{\partial z} = -\frac{i}{2} \longrightarrow \frac{\partial}{\partial z} = \frac{1}{2} \left(\frac{\partial}{\partial x} - i \frac{\partial}{\partial y} \right)$$

We note the following properties:

$$\frac{\partial}{\partial z} z = 1 \quad \frac{\partial}{\partial z} \bar{z} = \frac{1}{2} (1 - i^2) = 0$$

Interesting for is the result of the following:

$$\begin{aligned} \bar{z}^2 &= (x - iy)^2 = x^2 - y^2 - i2xy \\ \frac{\partial}{\partial z} \bar{z}^2 &= \frac{1}{2} \cdot (2x - i2y - i \cdot (-2y - i2x)) = x - iy + iy - ix = 0 \end{aligned}$$

4.3 Derivatives of the complex mean field equations

We can now compute derivatives of the complex functions $z(\mathbf{k}, t)$. We will set $z(\mathbf{k}, t) = z_{\mathbf{k}}$ and rewrite system (??) to help the reader:

$$\begin{aligned}\frac{\partial z_{\mathbf{k}}}{\partial t} &= -i \frac{(z_{\mathbf{k}} - 1)^2}{2} + \frac{(z_{\mathbf{k}} + 1)^2}{2} \cdot I_{\mathbf{k}} \quad z_{\mathbf{k}} \in \mathbb{C}^{M_{\mathbf{k}}} \\ I_{\mathbf{k}} &= -\Delta_{\mathbf{k}} + i\eta_{0_{\mathbf{k}}} + i\kappa \cdot H_{2_{\mathbf{k}}} \\ H_{2_{\mathbf{k}}} &= \frac{1}{\langle k \rangle} \sum_{\mathbf{k}'} P_{\mathbf{k}} a_{\mathbf{k}'\mathbf{k}} \cdot \left(1 + \frac{z_{\mathbf{k}'}^2 + \bar{z}_{\mathbf{k}'}^2}{6} - \frac{4}{3} \text{Re}(z_{\mathbf{k}'}) \right)\end{aligned}\tag{10}$$

The diagonal elements are found as:

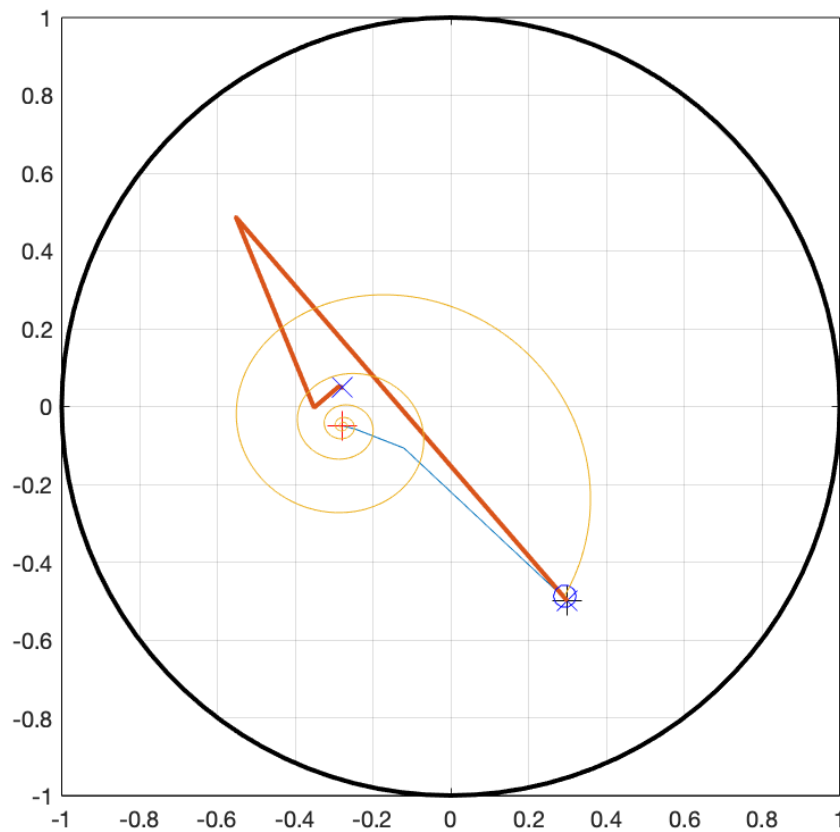
$$\begin{aligned}\frac{\partial}{\partial z_{\mathbf{k}}} \left(\frac{\partial z_{\mathbf{k}}}{\partial t} \right) &= -i(z_{\mathbf{k}} - 1) + (z_{\mathbf{k}} + 1) \cdot I_{\mathbf{k}} + \frac{(z_{\mathbf{k}} + 1)^2}{2} \cdot \frac{\partial I_{\mathbf{k}}}{\partial z_{\mathbf{k}}} \\ \frac{\partial I_{\mathbf{k}}}{\partial z_{\mathbf{k}}} &= i\kappa \cdot \frac{\partial H_{2_{\mathbf{k}}}}{\partial z_{\mathbf{k}}} \\ \frac{\partial H_{2_{\mathbf{k}}}}{\partial z_{\mathbf{k}}} &= \frac{1}{\langle k \rangle} P_{\mathbf{k}} a_{\mathbf{k}\mathbf{k}} \cdot \left(\frac{2z_{\mathbf{k}}}{6} - \frac{4}{3} \cdot \frac{1}{2} \right) = \frac{1}{\langle k \rangle} P_{\mathbf{k}} a_{\mathbf{k}\mathbf{k}} \cdot \frac{z_{\mathbf{k}} - 2}{3}\end{aligned}\tag{11}$$

The off-diagonal elements are found as

$$\begin{aligned}\frac{\partial}{\partial z_{\mathbf{k}'}} \left(\frac{\partial z_{\mathbf{k}}}{\partial t} \right) &= \frac{(z_{\mathbf{k}} + 1)^2}{2} \cdot \frac{\partial I_{\mathbf{k}}}{\partial z_{\mathbf{k}'}} \\ \frac{\partial I_{\mathbf{k}}}{\partial z_{\mathbf{k}'}} &= i\kappa \cdot \frac{\partial H_{2_{\mathbf{k}}}}{\partial z_{\mathbf{k}'}} \\ \frac{\partial H_{2_{\mathbf{k}}}}{\partial z_{\mathbf{k}'}} &= \frac{1}{\langle k \rangle} P_{\mathbf{k}'} a_{\mathbf{k}'\mathbf{k}} \cdot \frac{z_{\mathbf{k}'} - 2}{3}\end{aligned}\tag{12}$$

4.4 Results

I cannot seem to converge close enough.



5 Spike Timing Dependant Plasticity

Write about STDP, see the 'Postulates.pdf'.

With spike timing dependant plasticity we want to study how two neurons change their synaptic strength based on the time delay between spikes. When $\theta(t)_j > \pi$ we say that the neuron θ_j spikes at time t . Let us say that θ_i spikes at time t_i and θ_j spikes at t_j . Taking the time difference Δt_{ij} as $t_j - t_i$, we can say that when $\Delta t_{ij} > 0$ the spikes are correlated (there exists a temporally causal relation), and we can model an increase in synaptic strength of the connection K_{ij} from θ_i to θ_j . In the same fashion we can decrease K_{ji} when $\Delta t_{ij} < 0$ as there is no causal relation.

The functions $W(t)$ that relate Δt_{ij} to ΔK_{ij} are called *learning windows*, as they define a range in which K_{ij} is able to adapt. When signals between neurons show a very large time difference (negative or positive) we do not expect them to be correlated. Because the learning windows are generally not symmetrical we can also expect an asymmetrical adjacency matrix.

Another characteristic is the integral over the learning window. A window with a negative integral directs synaptic strengths mostly towards inhibitory behaviour, and vice versa with a positive integral. An integral of zero would mean that both inhibitory and excitatory synapses are stimulated equally.

Depending on the type of cell, different behaviour in time is

5.1 Biphasic Learning windows

5.1.1 Kempster 1999

$$W(t)_K = \eta \begin{cases} \left[A_p \left(1 - \frac{t}{\tilde{\tau}_p} \right) + A_n \left(1 - \frac{t}{\tilde{\tau}_n} \right) \right] \cdot \exp \left(\frac{t}{\tau_{\text{syn}}} \right) & \text{for } t \leq 0 \\ A_p \cdot \exp \left(-\frac{t}{\tau_p} \right) + A_n \cdot \exp \left(-\frac{t}{\tau_n} \right) & \text{for } t > 0 \end{cases} \quad (13)$$

Here t is the delay between presynaptic spike arrival and postsynaptic firing, η is a small learning parameter and all τ are time constants. Numerical values are usually $\eta = 0.05$, $\tau_{\text{syn}} = 5$ ms, $\tau_p = 1$ ms, $\tau_n = 20$ ms and $A_p = 1$ and $A_n = -1$. $\tilde{\tau}_p \equiv \tau_{\text{syn}}\tau_p/(\tau_{\text{syn}} + \tau_p)$ and $\tilde{\tau}_n \equiv \tau_{\text{syn}}\tau_n/(\tau_{\text{syn}} + \tau_n)$.
 $\int W(s)ds =$

5.1.2 Song 2000

The first formulation of *STDP* as a mathematical model was in [?].

$$W(t)_S = \begin{cases} A_p \cdot \exp \left(\frac{-t}{\tau_p} \right) & \text{for } s > 0 \\ A_n \cdot \exp \left(\frac{t}{\tau_n} \right) & \text{for } s \leq 0 \end{cases} \quad (14)$$

where we will use the values as proposed in [?]: $A_p = 0.1$, $A_n = -0.12$ and $\tau_p = \tau_n = 20$.

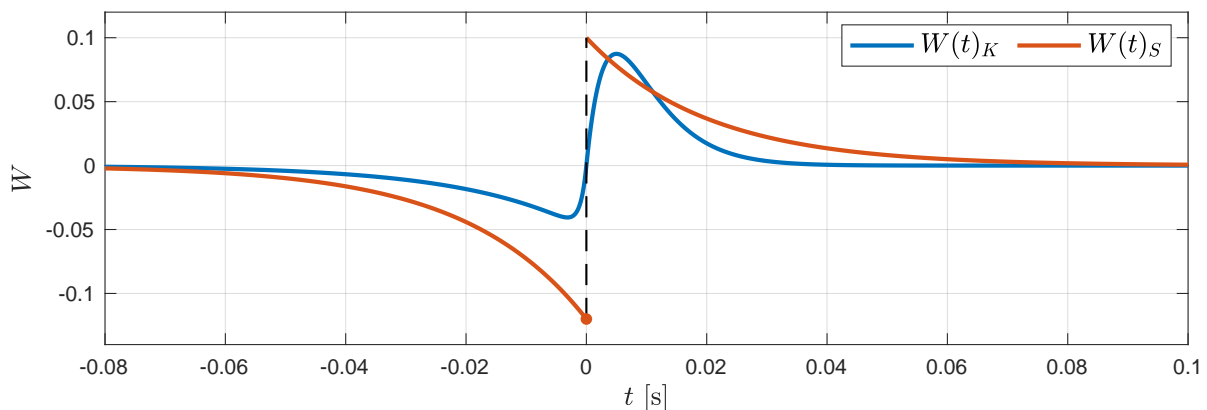


Figure 1: Two different biphasic learning windows. We can see how in $W(t)_s$ a larger weight is put on the anti-Hebbian learning.

5.2 Triphasic Learning windows

5.2.1 Chrol-Cannon 2012

$$W(t)_C = A_p \cdot \exp \left(\frac{-(t - 15)^2}{\tau_p} \right) - A_n \cdot \exp \left(\frac{-(t - 20)^2}{\tau_n} \right) \quad (15)$$

where $A_p = 0.23$, $A_n = 0.15$, $\tau_p = 200$ and $\tau_n = 2000$.

5.2.2 Waddington 2014

$$W(t)_W = A \left[1 - \frac{(t - \alpha)^2}{\alpha^2} \right] \cdot \exp \left(\frac{-|t - \alpha|}{\alpha} \right) \quad (16)$$

We will use $A = 0.1$ and $\alpha = 4.0$ ms . This brings the integral

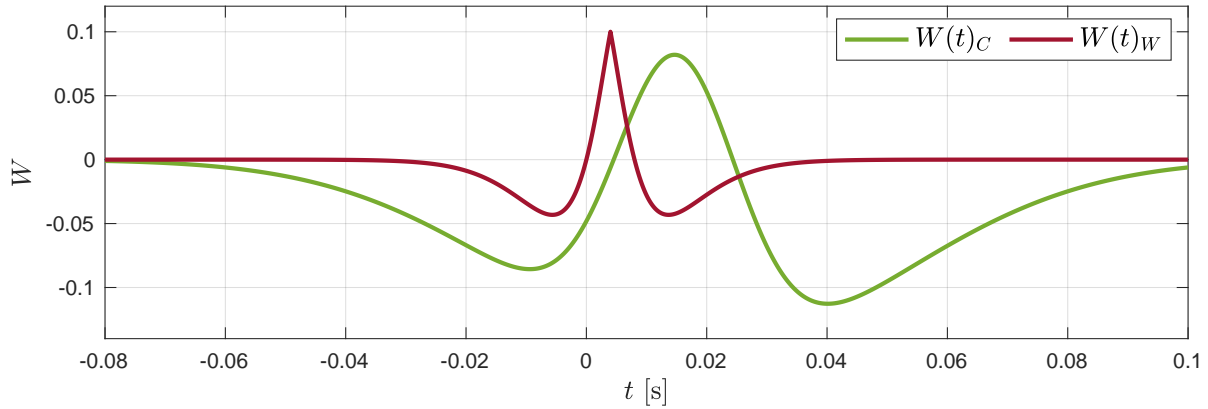


Figure 2: Two different triphasic learning windows. We can see how in $W(t)_C$ a much higher penalty is given to signals that arrive too early or too late. The moments at which signals should peak also differ by about 10 ms.

5.3 Custom learning windows

Some of the behaviour we want to observe is currently not accounted for: synaptic strengths are unbounded and are generally nonzero. How can we model the behaviour where synaptic strengths can also settle on being zero? Perhaps a better idea than updating the synaptic strength by adding a new value, we can scale it.

- When Δt_{ij} is very large (both positive and negative) we expect no change in the synaptic strength: $W(-\infty) = W(\infty) = 0$.
- We expect a specific positive time delay to yield the most amount of synaptic strengthening:

$$\Delta t_{\text{best}} = \arg \max_x W(t)$$