

# Derivation of the Effective Connectivity $J_{\text{eff}}(x, a, b)$

This note derives the Jacobian of the  $x$ -dynamics with respect to  $x$ , treating the adaptation variables  $a$  and the synaptic depression variables  $b$  as constants (i.e., “frozen” at a given time  $t_1$ ). The result is an effective connectivity matrix  $J_{\text{eff}}(x, a, b)$  that depends on the current state. Rather than considering the entire Jacobian in which the adaptation variables are included in the state, this approach helps the reader visualize how adaptation can be viewed as modulating connectivity, which is a concept investigated in the manuscript.

---

## 1. Model definition

---

We consider a network of  $N$  neurons (indices  $i, j = 1, \dots, N$ ) with the following equations:

### 1.1 $x$ -dynamics

The membrane / rate state variable  $x_i$  evolves as

$$\frac{dx_i}{dt} = \frac{-x_i + u_i + \sum_{j=1}^N w_{ij} r_j}{\tau_d},$$

where

- $\tau_d$  is the (global) decay time constant,
- $u_i$  is an external input (ignored when computing the Jacobian),
- $w_{ij}$  is the (structural) weight from neuron  $j$  to neuron  $i$ ,
- $r_j$  is the firing rate of neuron  $j$ .

### 1.2 Nonlinearity and adaptation

The firing rate  $r_i$  is given by

$$r_i = b_i \phi \left( x_i - a_{0i} - c \sum_{k=1}^K a_{ik} \right),$$

where

- $\phi(\cdot)$  is a static nonlinearity (e.g., sigmoid),
- $b_i$  is a synaptic depression / gain factor,
- $a_{0i}$  is an offset parameter (e.g., preferred input or threshold shift),
- $c$  is an adaptation gain,
- $a_{ik}$  are adaptation state variables for neuron  $i$  and component  $k = 1, \dots, K$ .

The dynamics of  $a_{ik}$  and  $b_i$  are not needed to compute  $J_{\text{eff}}$ , since we explicitly treat  $a$  and  $b$  as constants during this derivation.

---

## 2. Vector-matrix notation

---

Let

- $x = (x_1, \dots, x_N)^T$ ,
- $r = (r_1, \dots, r_N)^T$ ,
- $u = (u_1, \dots, u_N)^T$ ,
- $b = (b_1, \dots, b_N)^T$ ,
- $W = (w_{ij}) \in \mathbb{R}^{N \times N}$ ,
- $1$  be the  $N$ -vector of ones.

The recurrent input term can be written as

$$\left( \sum_{j=1}^N w_{ij} r_j \right)_i = (Wr)_i.$$

To make it explicit that  $r_j$  scales the columns of  $W$ , we can also write

$$Wr = W\text{diag}(r) \mathbf{1},$$

since

$$(W \text{diag}(r) \mathbf{1})_i = \sum_j w_{ij} r_j.$$

Thus, the  $x$ -dynamics in vector form is

$$\dot{x} = \frac{-x + u + Wr}{\tau_d} = \frac{-x + u + W \text{diag}(r) \mathbf{1}}{\tau_d}.$$

---

### 3. Jacobian of $r$ with respect to $x$

---

We treat  $a_{ik}$  and  $b_i$  as constants. For each neuron  $i$ ,

$$r_i = b_i \phi \left( x_i - a_{0i} - c \sum_{k=1}^K a_{ik} \right).$$

Define the constant

$$\text{const}_i = a_{0i} + c \sum_{k=1}^K a_{ik}.$$

Then

$$r_i = b_i \phi(x_i - \text{const}_i).$$

The partial derivative of  $r_i$  with respect to  $x_j$  is

$$\frac{\partial r_i}{\partial x_j} = b_i \phi'(x_i - \text{const}_i) \delta_{ij},$$

where  $\delta_{ij}$  is the Kronecker delta.

Define the gain vector

$$g_i = b_i \phi' \left( x_i - a_{0i} - c \sum_{k=1}^K a_{ik} \right),$$

and the corresponding diagonal matrix

$$G = \text{diag}(g_1, \dots, g_N).$$

In matrix form, the Jacobian of  $r$  with respect to  $x$  is

$$\frac{\partial r}{\partial x} = G.$$

---

#### 4. Jacobian of $\dot{x}$ with respect to $x$

---

We now compute

$$J_{\text{eff}}(x, a, b) = \frac{\partial \dot{x}}{\partial x}.$$

Recall

$$\dot{x} = \frac{-x + u + Wr}{\tau_d}.$$

#### 4.1 Contribution from the leak term

For the leak term  $-x/\tau_d$ ,

$$\frac{\partial}{\partial x_j} \left( -\frac{x_i}{\tau_d} \right) = -\frac{1}{\tau_d} \delta_{ij}.$$

Thus, in matrix form this contributes

$$-\frac{1}{\tau_d} I,$$

where  $I$  is the  $N \times N$  identity matrix.

#### 4.2 Contribution from the recurrent term

For the recurrent term  $(1/\tau_d)Wr$ ,

$$\frac{\partial}{\partial x_j} \left( \frac{1}{\tau_d} \sum_{k=1}^N w_{ik} r_k \right) = \frac{1}{\tau_d} \sum_{k=1}^N w_{ik} \frac{\partial r_k}{\partial x_j}.$$

Using  $\partial r_k / \partial x_j = g_k \delta_{kj}$ , we get

$$\frac{\partial}{\partial x_j} \left( \frac{1}{\tau_d} \sum_{k=1}^N w_{ik} r_k \right) = \frac{1}{\tau_d} w_{ij} g_j.$$

So the entries of the Jacobian are

$$J_{\text{eff},ij} = \frac{\partial \dot{x}_i}{\partial x_j} = -\frac{1}{\tau_d} \delta_{ij} + \frac{1}{\tau_d} w_{ij} g_j.$$

In matrix form, note that  $(WG)_{ij} = \sum_k w_{ik} G_{kj} = w_{ij} g_j$ , since  $G$  is diagonal. Therefore

$$J_{\text{eff}}(x, a, b) = \frac{1}{\tau_d} (-I + WG).$$

Explicitly,

$$G = \text{diag}\left(b_i \phi'(x_i - a_{0i} - c \sum_{k=1}^K a_{ik})\right).$$

---

## 5. Summary

---

- The effective connectivity Jacobian for the  $x$ -dynamics, treating  $a$  and  $b$  as constants and ignoring external input  $u$  in the derivative, is

$$J_{\text{eff}}(x, a, b) = \frac{1}{\tau_d} (-I + WG),$$

where

$$G = \text{diag}\left(b_i \phi'(x_i - a_{0i} - c \sum_{k=1}^K a_{ik})\right).$$

- Entrywise, this is

$$J_{\text{eff},ij} = -\frac{1}{\tau_d} \delta_{ij} + \frac{1}{\tau_d} w_{ij} b_j \phi' \left( x_j - a_{0j} - c \sum_{k=1}^K a_{jk} \right).$$