

Optimizing individually the variance of the neural activity regulates the echo-state spectral radius

Fabian Schubert and Claudius Gros

Institute for Theoretical Physics, Goethe University, Frankfurt a.M., Germany.

Keywords: variance optimization, echo state network, spectral radius, biological plausibility, self-organization, universality

1 Introduction

Strategies for the optimization of ESN hyperparameters can be divided in two categories: supervised and unsupervised methods, where the first one utilizes an error signal, while the latter only uses information contained within the network dynamics.

In the first part of our research, we investigated the possibility of defining an unsupervised homeostatic mechanism that controls the mean and variance of neuronal firing in such a way that the network acts in a regime that yields good performance in sequence learning tasks. This mechanism acts on two sets of parameters, biases b_i and neural gain factors a_i . It should be emphasized that we did not attempt to define an arbitrarily complex measure that would be most suitable for optimization, e.g. from a machine learning perspective. Rather, we restricted ourselves to adhere to biologically plausible mechanisms. While no exact definition of this term exists, it embraced two aspects in our work:

- The dynamics of all variables must be local, i.e., they are bound to a specific neuron and may only access other variables that are locally accessible. In a strict sense, this consists of all other dynamic variables of the neuron itself and information about the activity of adjacent neurons. Being less restrictive, one could claim that it should also be possible to access aggregate or mean-field quantities, that average a property over the entire population.
- We use a time-discrete model where the state of a variable in the next step may only be determined by states of the previous step. This means that information about past states must be integrated dynamically.

Our approach was based on the assumption that network performance is optimal when the spectral radius of the effective recurrent connectivity, given by $a_i W_{ij}$, is close

Table 1: Standard network parameters.

N	D_{in}	D_{out}	p_r	σ_w
500	1	1	0.1	1

to, but slightly below 1. We attempted to transfer this non-local measure into a condition that could be implemented in a biologically plausible way.

2 Model

2.1 Network dynamics

$$y_i(t) = \tanh(x_i(t) - b_i) \quad (1)$$

$$x_i(t) = X_{r,i}(t) + X_{e,i}(t) \quad (2)$$

$$X_{r,i}(t) = a_i \sum_{j=1}^N W_{ij} y_j(t-1) \quad (3)$$

$$X_{e,i}(t) = \sum_{j=1}^{D_{\text{in}}} W_{ij}^u u_j(t) \quad (4)$$

$$o_i(t) = o_i^0 + \sum_{j=1}^{D_{\text{out}}} W_{ij}^o y_j(t) \quad (5)$$

where $\mathbf{y}, \mathbf{x}, \mathbf{a}, \mathbf{b} \in \mathbb{R}^N$, $W \in \mathbb{R}^{N \times N}$, $\mathbf{u} \in \mathbb{R}^{D_{\text{in}}}$, $W^u \in \mathbb{R}^{N \times D_{\text{in}}}$, $\mathbf{o}, \mathbf{o}^0 \in \mathbb{R}^{D_{\text{out}}}$ and $W^o \in \mathbb{R}^{D_{\text{out}} \times N}$.

Furthermore

$$p(W_{ij} = x) = \begin{cases} \delta(x) & i = j \\ p_r \mathcal{N}(x, \mu = 0, \sigma = \sigma_w / \sqrt{N p_r}) + (1 - p_r) \delta(x) & \text{else} \end{cases} \quad (6)$$

$$p(W_{ij}^u = x) = \mathcal{N}(x, \mu = 0, \sigma = 1) \quad (7)$$

Initially, we chose $N = 1000$ as the network size, however, due to computational complexity, the results presented here are generated with a network of size $N = 500$, unless stated otherwise. See Table 1 for the standard network parameters.

2.2 Homeostatic Adaptation

For our homeostatic update mechanism, we use the following dynamics:

$$b_i(t) = b_i(t-1) + \epsilon_b [y_i(t) - \mu_i^t] \quad (8)$$

$$\mu_i^y(t) = [1 - \epsilon_\mu] \mu_i^y(t-1) + \epsilon_\mu y_i(t) \quad (9)$$

$$a_i(t) = a_i(t-1) + \epsilon_a [\sigma_i^{t^2} - (y_i(t) - \mu_i^y(t))^2] \quad (10)$$

See Table 2 for the standard values.

Table 2: Standard homeostasis parameters.

ϵ_b	ϵ_μ	ϵ_a	μ_i^t	σ_i^t
10^{-3}	10^{-4}	10^{-3}	0.05	variable

3 Theory

We stated in the model description that all recurrent weights were drawn independently from a sparse Gaussian distribution. In some sense, of course, this is an assumption that already fulfills one of the conditions that is known to make ESNs work, namely a balance between excitation and inhibition. However, since adjusting gains does not provide a means to dynamically achieve this property, we had to take it as given.

Furthermore, we assumed that all entries of the weight matrix were independently drawn from the same distribution. Under this assumption, using the circular law, it follows that setting all gain values to $1/\sigma_w$ will result in a uniform unit circle distribution of eigenvalues. The situation is less clear if we allow the rows or columns of the matrix to follow different distributions, in particular—since we assumed to have zero mean—different variances. Numerically, we found that if

$$R_a \equiv \langle R_{a,i} \rangle_P = \left\langle a_i^2 \sum_j W_{ij} \right\rangle_P = 1 \quad (11)$$

where $\sigma_{w,i}^2$ are the variances of individual rows of the matrix, the spectral radius will be unity. This observation was also reported and proven in (Rajan and Abbott, 2006).

From this observation, two questions emerge: first, can we find local gain dynamics that can tune the global measure R_a ? And second, given the assumption that this constraint is fulfilled, does the particular distribution of gains affect the network performance?

Since R_a is already a population average itself, we could simply argue that a biologically plausible mechanism that adjusts gains proportionally would be

$$a_i(t+1) = a_i(t) + \epsilon_a a_i(t) [R_a^t - R_a(t)] \quad (12)$$

where R_a^t is the spectral radius we would like to achieve.

The question at hand is: could a diffusive neuromodulator encode the squared product of the neural gain and the variance of its synaptic weights? Or, alternatively, could this quantity be encoded in a different physical variable?

3.1 Spectral Radius Encoded in Input and Neural Activity

One possible way to encode the aforementioned quantity in physical way is based on the very general statement that each configuration of gains, weights and input statistics will yield a particular set of statistics for the resulting neural activity. Therefore, we hypothesized that it should be possible to establish an (approximate) description of the causal relation of these properties. In particular, this would potentially allow us to link a given value of R_a , expressed as a function of gains and weights, to a corresponding set of input and neural activity statistics.

Based on this approach, we came up with an approximate self-consistency equation that was based on the following assumptions/simplifications:

- We reduced the set of gains a_i to single variable a .
- Every neuron receives statistically independent external input, which is however drawn from the same statistic characterized by the standard deviation σ_{ext} . We assumed the mean input to be zero.
- Each neurons has the same homeostatic target variance.
- We ignore cross-correlations of activity in the neural population. Furthermore, we also assumed that each neuron has zero autocorrelation for $\tau \neq 0$.
- Even though the external input signal could in theory follow an arbitrary statistical distribution, we model the statistics of the sum of recurrent and external input with a Gaussian distribution.
- We ignore the small offset μ_i^t in the mean activity.

Under these assumptions, the networks fulfills a self-consistency equation given by

$$\sigma_t^2 = \int_{-\infty}^{\infty} dx \tanh^2(x) N_{\mu, \sigma}(x), \quad \mu = 0, \quad \sigma^2 = a^2 \sigma_w^2 \sigma_t^2 + \sigma_{\text{ext}}^2, \quad (13)$$


where the distribution $N_{\mu, \sigma}(x)$ is of the membrane potential x is a Gaussian, with mean $\mu = 0$ and variance σ^2 . The gain a is adapted such that (13) is fulfilled, defining a 2d-manifold in the $(\sigma_t, \sigma_{\text{ext}}, a)$ space.

As a ‘sanity check’, we ran a network simulation where the second and fourth approximation was actually exact. That is, we used Gaussian white noise with zero mean and σ_{ext} standard deviation for all neurons. Furthermore, the target mean was set to zero. Running the network for different pairs of external input variance and target variance, we got the results shown in Fig. 1. The theoretical prediction given by (13) provides a good estimate for the population average of the squared gains.

After this, we relaxed the assumptions and tested combinations of three variants: first, by introducing a variation in input variances by drawing input weights from a normal distribution whose variance was given by σ_{ext}^2 , see Fig. 10. Second, instead of independent Gaussian driving, we fed an identical binary input into every node, but input weights were the same, see Fig. 11. Third, we tested a combination of both previous variants, see Fig. 12.

Comparing the results of these simulations, we can generally state that the quality of the gain depends on the amount of error within the prediction of the recurrent membrane variance: while errors are quite low in the autonomous case, they can be significantly increased (Fig. 11 and 12) or further decreased (Fig. 1 and 10), depending on the statistical properties of the external driving.

In Echo state networks, the best task performance is usually found for a spectral radius close to unity. Therefore, our idea was to find an implicit expression for the spectral radius being one. As shown in Fig. 1, 10, 11 and 12, for a given external driving there exists a unique target activity variance that corresponds to a spectral radius of one.



`./plots/homogeneous_independent_gaussian_input_compos.pdf`

Figure 1: Homogeneous Independent Gaussian Input. **A**: Population average of inter-neuron cross correlation. **B**: Prediction error of recurrent input variance. **C**: Theoretical prediction of R_a vs. numerical results.

./plots/homogeneous_independent_gaussian_input_rec_mem_pot_predict_size.

Figure 2: Squared Recurrent Input Variance Prediction Error for homogeneous independent Gaussian input.

Before proceeding to an analytical approach to this idea, we ran a numeric test by running a parameter sweep over pairs of $\sigma_t, \sigma_{\text{ext}}$, adhering to the most simplified version, where each neuron received independent Gaussian external input with the same variance. The resulting transition line is shown in Fig. 3. Finding an analytic expression for this line would allow tuning the network into this transition point by measuring external and recurrent input variance. The following section explains an approximation method for finding an expression for this transition.

3.2 Gaussian Approximation

A polynomial approximation of \tanh^2 to fourth order captures the right behavior close to the origin, but does not account for the fact that \tanh^2 converges to 1 for large absolute values of the membrane potential. Alternatively, an approximation with the correct scaling to second order as well as the right convergence is

$$\tanh^2(x) \approx 1 - \exp(-x^2) . \quad (14)$$

Using this approximation in (13) and solving the integral yields

$$\sigma_t^2 = 1 - 1/\sqrt{1 + 2a^2\sigma_w^2\sigma_t^2 + 2\sigma_{\text{ext}}^2} . \quad (15)$$

Solving this equation for a gives

$$a = \sigma_w^{-1} \sqrt{\frac{1 - (1 - \sigma_t^2)^2 (1 + 2\sigma_{\text{ext}}^2)}{2(1 - \sigma_t^2)^2 \sigma_t^2}} . \quad (16)$$

We can also derive an approximation for the critical transition from (15), which is given by

$$\sigma_{\text{ext}} = \sqrt{\frac{1}{2(1 - \sigma_t^2)^2} - \sigma_t^2 - \frac{1}{2}}. \quad (17)$$

See Fig. 3 for a comparison. One can obtain an even simpler approximation by

$$\sigma_{\text{ext}} = \sqrt{\frac{1}{2(1 - \sigma_t^2)^2} - \sigma_t^2 - \frac{1}{2}} \quad (18)$$

$$= \sqrt{2}^{-1} \sqrt{\frac{1}{(1 - \sigma_t^2)^2} - 2\sigma_t^2 - 1} \quad (19)$$

$$= \sqrt{2}^{-1} \frac{\sqrt{3\sigma_t^4 - 2\sigma_t^6}}{1 - \sigma_t^2} \quad (20)$$

$$\approx \sqrt{\frac{3}{2}} \frac{\sigma_t^2}{1 - \sigma_t^2} \quad (21)$$

where we ignored the sixth order term. (21) is also shown in Fig. 3 for comparison. This simple form also allows us conveniently state the inverse relation, which is what is required for tuning the network activity into an optimal state:

$$\sigma_t \approx 1 / \sqrt{\sqrt{\frac{3}{2}} \sigma_{\text{ext}}^{-1} + 1}. \quad (22)$$

However, this result is only valid for the particular case described: independent homogeneous Gaussian input. As Fig. 10 – 12 indicate, the manifold of the set of solutions can significantly change its shape, depending on particular input statistics—heterogeneity in the external input variance apparently being the dominant factor. In principle, this heterogeneity could be accounted for by modifying (15) to only represent a local per-node condition:

$$\sigma_{t,i}^2 = 1 - 1 / \sqrt{1 + 2a_i^2 \sigma_{w,i}^2 \langle \sigma_{t,i}^2 \rangle_P + 2\sigma_{\text{ext},i}^2}. \quad (23)$$

Averaging over this yields a much more complex self-consistency equation:

$$\langle \sigma_{t,i}^2 \rangle_P = 1 - \left\langle 1 / \sqrt{1 + 2a_i^2 \sigma_{w,i}^2 \langle \sigma_{t,i}^2 \rangle_P + 2\sigma_{\text{ext},i}^2} \right\rangle_P. \quad (24)$$

In this more general case, the critical transition would be defined by $\langle a_i^2 \sigma_{w,i}^2 \rangle_P = 1$. Apart from the difficulties of finding the set of possible variance solutions for this condition, it is evident that it would depend on information about all external input variances, violating the assumptions we made about biological plausibility, even if we restrict ourselves to the more uniform case of $a_i^2 \sigma_{w,i}^2 = 1, \forall i$.

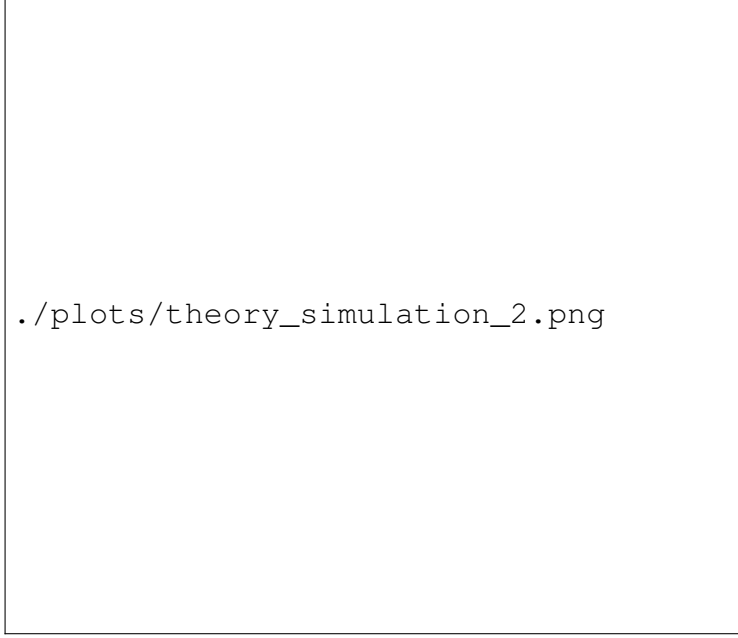


Figure 3: **Spectral Radius Transition Approximation** Blue dashed lines denotes $\langle a_i^2 \rangle_P = 1/\sigma_w^2$. Yellow dashed line is given by (21), dotted line by (17).

4 XOR-Memory Recall

To test the whether our self-consistent analytical approach would correctly identify the critical transition under some task-related input, we rand a sweep search over a range of values for σ_{ext} and σ_t , using a binary sequence as input. After the initial adaptation, the network's performance was tested with an XOR-recall task defined by

$$f(t) = \text{XOR}[u(t - \tau), u(t - \tau - 1)] \quad (25)$$

$$\text{XOR}[x, y] := \begin{cases} 0 & x = y \\ 1 & \text{else} \end{cases} \quad (26)$$

where $u(t) \in \{-1, 1\}$. The performance was quantified by the total XOR memory capacity given by

$$\text{MC}_{\text{XOR}} := \sum_{k=1}^{\infty} \text{MC}_{\text{XOR},k} \quad (27)$$

$$\text{MC}_{\text{XOR},k} := \max_{w_{\text{out}}} \frac{\text{Cov}^2[\text{XOR}[u(t - k), u(t - k - 1)], y_{\text{out}}(t)]_t}{\text{Var}[\text{XOR}[u(t), u(t - 1)]]_t \text{Var}[y_{\text{out}}(t)]_t} . \quad (28)$$

Fig. 4 depicts the performance results for this setup. Two different schemes were tested: The first one used independent Gaussian noise for the external input of each node during adaptation, where the standard deviation of the distribution was homogeneous across the population. For the actual performance measurement, a binary input sequence was fed into the network, using Gaussian distributed input weights that resulted in an average

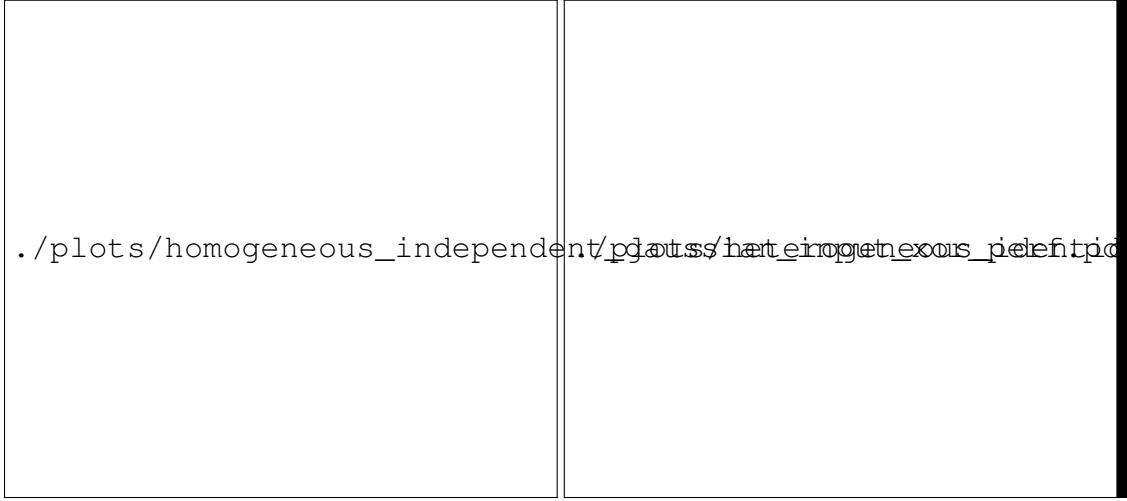


Figure 4: XOR-memory performance as defined in (27). Left: Network adapted with homogeneous independent Gaussian input, tested with $\{-1, 1\}$ binary input with the same standard deviation. Right: Network adapted with heterogeneous identical binary input, tested with the same input type. Blue dashed line: numerically determined points where the spectral radius is one. Orange line: maximal performance for a given σ_{ext} . Green line: theoretical prediction for unit spectral radius.

σ_{ext} of the same magnitude as was used during adaptation. In the second case, the same type of binary input was also used during adaptation. While the first approach yielded a good match between the expected shape of the critical transition line and the numerical result, optimal performance under a given external driving did not closely follow this transition. The second case (using only binary input) did exhibit a better correspondence between optimal performance and the critical transition. However, our theoretical predictions did not match this transition anymore.

5 A scaling mechanism estimating the phase space contraction rate

As we have seen, the previous approach did not yield the desired results. However, we may take a slightly different approach to this problem by stating that the essential property that we are trying to reach is that a mapping from neural activities at time step $t - 1$ to the recurrent membrane potential at t should, on average, neither be contracting nor expanding. That is

$$\left\langle \frac{\|\mathbf{X}_r(t)\|^2}{\|\mathbf{y}(t-1)\|^2} \right\rangle_t = 1. \quad (29)$$

Being aware that we do not necessarily find exact equality, we can also demand that

$$\frac{\langle \|\mathbf{X}_r(t)\|^2 \rangle_t}{\langle \|\mathbf{y}(t-1)\|^2 \rangle_t} = 1 \quad (30)$$

$$\langle \|\mathbf{X}_r(t)\|^2 \rangle_t = \langle \|\mathbf{y}(t-1)\|^2 \rangle_t. \quad (31)$$

Note that previously we already assumed to have access to a mean field over the square of neural activities. If we assume that we can also access a mean field over recurrent membrane potentials, this allows us to modify the gain update rule in the following way:

$$X_{r,i}(t) = a_i(t) \sum_{j=1}^N W_{ij} y_j(t-1) \quad (32)$$

$$a_i(t) = a_i(t-1) + \epsilon_a a_i(t-1) [\|\mathbf{y}(t-1)\|^2 - \|\mathbf{X}_r(t)\|^2]. \quad (33)$$

The proportionality factor $a_i(t-1)$ is optional and simply accounts for a “stretching” of the gain vector proportional to the difference term. Importantly, if the adaptation rate ϵ_a is sufficiently slow, $\|\mathbf{y}(t-1)\|^2$ in (33) can be replaced by $\|\mathbf{y}(t)\|^2$:

$$a_i(t) = a_i(t-1) + \epsilon_a a_i(t-1) [\|\mathbf{y}(t)\|^2 - \|\mathbf{X}_r(t)\|^2]. \quad (34)$$

Fig. 6 and 7 show that in this implementation, the tuning towards unit spectral radius is almost perfect.

To better understand the dynamics of this mechanism, we approximate the average dynamics and reduce it to the case of a single scaling factor a and $\|\mathbf{y}(t)\|^2$:

$$\langle \Delta a \rangle_t = \epsilon_a a (1 - a^2 \sigma_w^2) \|\mathbf{y}(t)\|^2 \quad (35)$$

$$\langle \Delta \|\mathbf{y}\|^2 \rangle \approx N \left[1 - \left\langle \frac{1}{1 + 2a^2 \sigma_w^2 \|\mathbf{y}\|^2 / N + 2\sigma_{\text{ext},i}} \right\rangle_P \right] - \|\mathbf{y}\|^2 \quad (36)$$

These dynamics are depicted in Fig. 5. One stable fixed point corresponds to the desired property for the spectral radius.

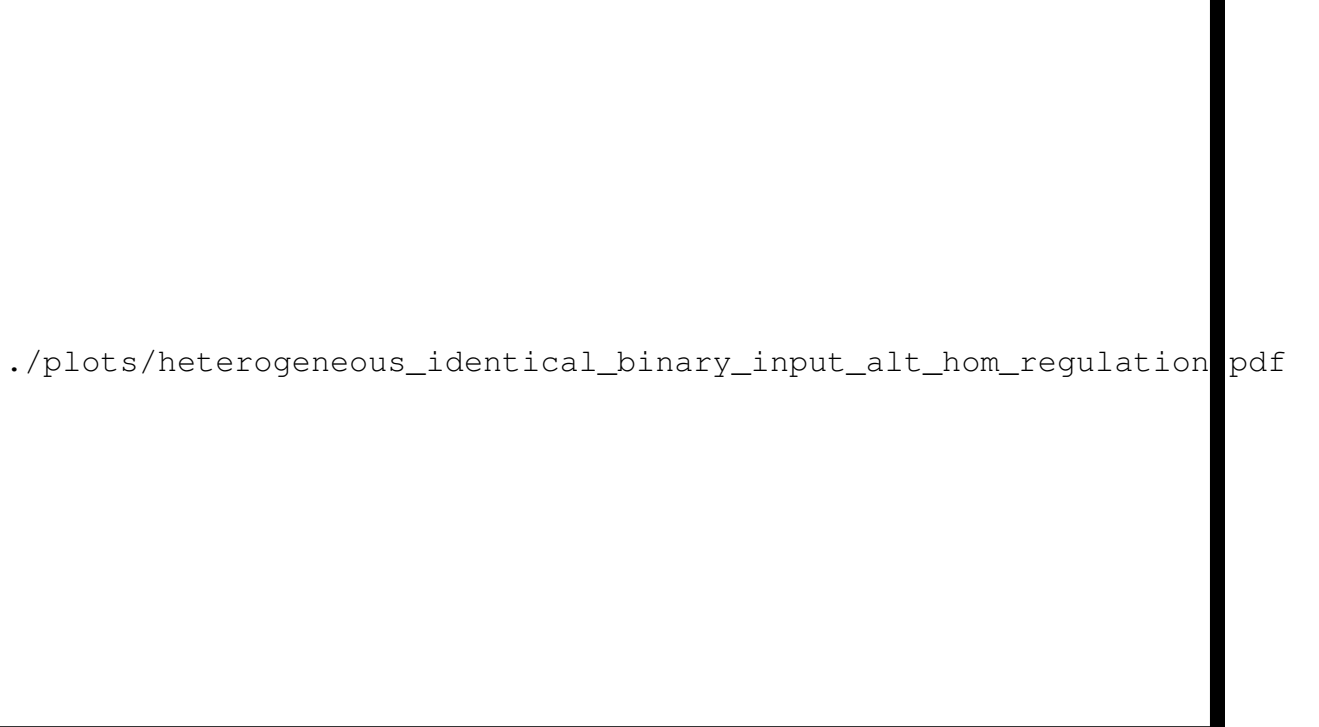


Figure 5: Dynamics of gain and squared activity as described in (35) and (36). Shown is the flux given by the equations (blue), as well as traces of actual network/gain dynamics (orange), where the latter is given by (34). Green and red lines depict nullclines of gain and squared activity, respectively. Input to the network was heterogeneous identical binary input. $\sigma_{\text{ext}} = 0.05$, $\epsilon_a = 0.1$.



Figure 6: Gain dynamics as described in section 5, using heterogeneous identical binary input.



Figure 7: Effective eigenvalues before and after adaptation as described in section 5, using heterogeneous identical binary input.

Conclusion

We have illustrated the basic format to the manuscript that you consider to submit to Neural Computation. We hope this is helpful to the authors.

Acknowledgments

The people you want to acknowledge. For this document, we appreciate Jrg Lcke, author of an accepted paper who generously allowed us to use his template.

Appendix

A Gradient-Based Error-Driven Adaptation

Instead of trying to find some heuristics that would tune the network into a state that is supposed to be “generally suitable” for time dependent problems / series prediction, we could adapt our parameters based on an error signal. Suppose we have an error between an output signal $u(t)$ and a target signal $f(t)$, we can define the error measure

$$\epsilon(t) = u(t) - f(t) \quad (37)$$

$$E(t) = \frac{1}{2}\epsilon(t)^2 \quad (38)$$

$$\mathcal{L} = \langle E(t) \rangle_t . \quad (39)$$

A gradient-based approach to reducing E with respect to a_i would read

$$\Delta a_i \propto -\frac{d}{da_i}\mathcal{L} = -\left\langle \frac{d}{da_i}E(t) \right\rangle_t \quad (40)$$

One way to evaluate this expression is called *backpropagation through time* (BPTT). Essentially, BPTT unfolds the gradient into the following sum:

$$\frac{d}{da_i}\mathcal{L} = \sum_{t=-\infty}^{\infty} \frac{d\mathcal{L}}{d\mathbf{y}(t)} \frac{\partial \mathbf{y}(t)}{\partial a_i} \quad (41)$$

Partial derivatives with respect to neural activities are calculated for each time step and the full derivatives of the error with respect to these activities can be subsequently calculated via the chain rule, going “backwards” in time. This method does requires the evaluation of $\frac{d\mathcal{L}}{d\mathbf{y}(t)}$ to be truncated at some point, obviously. An alternative, known as *real time recurrent learning* (RTRL) calculates full derivatives of activities with respect to the parameter, but only partial derivatives of the error measure with respect to these activities:

$$\frac{d}{da_i}\mathcal{L} = \sum_{t=-\infty}^{\infty} \frac{\partial \mathcal{L}}{\partial \mathbf{y}(t)} \frac{d\mathbf{y}(t)}{da_i} \quad (42)$$

the product between the two vectors has to be interpreted as a scalar product. Mathematically, these expressions are exactly equivalent, but in practice, lead to different implementations. We can see this more clearly by plugging in the definition of the error measure.

BPTT:

$$\frac{d}{da_i} \mathcal{L} = \sum_{t=-\infty}^{\infty} \frac{d \langle E(t') \rangle_{t'}}{d\mathbf{y}(t)} \frac{\partial \mathbf{y}(t)}{\partial a_i} \quad (43)$$

$$= \lim_{T \rightarrow \infty} \frac{1}{2T+1} \sum_{t'=-T}^T \sum_{t=-\infty}^{\infty} \frac{dE(t')}{d\mathbf{y}(t)} \frac{\partial \mathbf{y}(t)}{\partial a_i} \quad (44)$$

$$(45)$$

Since our model does not contain causation backwards in time, we can cut the sum by

$$\frac{d}{da_i} \mathcal{L} = \lim_{T \rightarrow \infty} \frac{1}{2T+1} \sum_{t'=-T}^T \sum_{t=-\infty}^{t'} \frac{dE(t')}{d\mathbf{y}(t)} \frac{\partial \mathbf{y}(t)}{\partial a_i} \quad (46)$$

In the case of stochastic gradient descent, this is split up into a series of $\Delta a_i(t)$ that is subsequently added to the parameter where

$$\Delta a_i(t') = -\epsilon_a \sum_{t=-\infty}^{t'} \frac{dE(t')}{d\mathbf{y}(t)} \frac{\partial \mathbf{y}(t)}{\partial a_i}. \quad (47)$$

Furthermore, as mentioned before, this sum has to be truncated at some point:

$$\Delta a_i(t') \approx -\epsilon_a \sum_{t=t'-t_{\text{trunc}}}^{t'} \frac{dE(t')}{d\mathbf{y}(t)} \frac{\partial \mathbf{y}(t)}{\partial a_i}. \quad (48)$$

Proceeding in the same way with (42), we get

$$\frac{d}{da_i} \mathcal{L} = \lim_{T \rightarrow \infty} \frac{1}{2T+1} \sum_{t'=-T}^T \sum_{t=-\infty}^{\infty} \frac{\partial E(t')}{\partial \mathbf{y}(t)} \frac{d\mathbf{y}(t)}{da_i} \quad (49)$$

Similar to the causal argument in BPTT, we find that $\frac{\partial E(t')}{\partial \mathbf{y}(t)}$ is only nonzero if $t = t'$, which allows us to drop the inner sum entirely:

$$\frac{d}{da_i} \mathcal{L} = \lim_{T \rightarrow \infty} \frac{1}{2T+1} \sum_{t=-T}^T \frac{\partial E(t)}{\partial \mathbf{y}(t)} \frac{d\mathbf{y}(t)}{da_i} \quad (50)$$

Again, stochastic gradient descent splits up this sum into subsequent changes:

$$\Delta a_i(t) = -\epsilon_a \frac{\partial E(t)}{\partial \mathbf{y}(t)} \frac{d\mathbf{y}(t)}{da_i} \quad (51)$$

While this might seem like a simpler expression than (48), the challenge of this approach is to evaluate the *full* derivative $\frac{d\mathbf{y}(t)}{da_i}$ in each time step. Practically, this is done in an iterative way:

$$\frac{d\mathbf{y}(t)}{da_i} = \frac{\partial \mathbf{y}(t)}{\partial a_i} + \left[\frac{\partial \mathbf{y}(t)}{\partial \mathbf{y}(t-1)} \right] \frac{d\mathbf{y}(t-1)}{da_i}. \quad (52)$$

Note that the expression $\frac{\partial \mathbf{y}(t)}{\partial \mathbf{y}(t-1)}$ is a rank two tensor where $\left[\frac{\partial \mathbf{y}(t)}{\partial \mathbf{y}(t-1)} \right]_{ij} = \frac{\partial y_i(t)}{\partial y_j(t-1)}$. A problem with this iterative approach is that, if used in combination with stochastic gradient descent, a_i will change while iteratively updating the derivatives. This means that derivatives are only exact in the “adiabatic” limit of very small learning rates.

A.1 Biological Plausibility in Backpropagation


The question how a backpropagation scheme could be implemented in the brain is an ongoing research field (see e.g. Whittington and Bogacz (2019)). A particular variant of this question is how causal relations over time in a recurrent network could be learned in a biologically plausible fashion. Comparing the previously discussed frameworks, it appears to be the case that RTRL is a better candidate for this question: In each instance time instance, the update rule only consists of two terms that can be interpreted rather easily: A term that accounts for the effect of neural activities on the error signal, and a term that expresses the effects of changes in the actual parameters onto these activities. Some recent approaches to biologically plausible learning (see Murray (2019); Bellec et al. (2019)) have essentially proposed to drop the second iterative term in (52) to avoid the non-local nature of this equation. Written out in complete form, this yields

$$\Delta a_i(t) = -\epsilon_a \epsilon(t) W_i^0 [1 - y_i^2(t)] \sum_j W_{ij} y_j(t-1). \quad (53)$$

We compared this approximate learning rule for the XOR-recall task with the full RTRL rule by stochastically estimating the gradients. In Fig. 8 we reduced the dynamics to a global gain parameter (by taking the mean over individual gain gradients) for illustration purposes and also included the actual error landscape from which the gradient was to be derived.

The approximate gradients are following quite closely. However, we can see that if the recall length becomes longer, an additional minimum at zero gain occurs that the gradient based adaptation could potentially end up in.

We further investigated the effect of the network size onto the shape of the objective function, see Fig. 9. As expected, performance generally increased (MSE decreased) monotonically with increasing network size. Furthermore, larger networks can increase the width of the optimal parameter range, as seen for $\tau = 5$. Another visible effect of decreasing network size is the disappearance of local minima, eventually leading to the aforementioned global minimum at zero gain. It appears that this “catastrophic shutdown” of gains in case no good solution can be found is less of an issue for large networks. Still, since it is the optimal solution if the network can not succeed in extracting actual information from external driving (i.e., “it is better to say nothing than to



`./plots/delta_a_local.pdf`

Figure 8: Error gradients calculated from full RTRL and from the approximation given in (53). Mean square error (MSE) is shown in green.

have a random guess”), it seems likely that if an actual biological system was subject to error driven adaptation, it would also use some form of homeostatic regulation to avoid this effect.

B Additional Figures



`./plots/performance_network_size.pdf`

Figure 9: Mean square error for the XOR-recall task as a function of gain for different network sizes.


References

- Bellec, G., F. Scherr, E. Hajek, D. Salaj, R. Legenstein, and W. Maass
2019. Biologically inspired alternatives to backpropagation through time for learning in recurrent neural nets.
- Murray, J. M.
2019. Local online learning in recurrent networks with random feedback. *eLIFE*.
- Rajan, K. and L. F. Abbott
2006. Eigenvalue spectra of random matrices for neural networks. *Physical Review Letters*, 97(18).
- Whittington, J. C. and R. Bogacz
2019. Theories of Error Back-Propagation in the Brain. *Trends in Cognitive Sciences*, 23(3):235–250.




`./plots/heterogeneous_independent_gaussian_input_compos.pdf`

Figure 10: Heterogeneous Independent Gaussian Input. **A**: Population average of inter-neuron cross correlation. **B**: Prediction error of recurrent input variance. **C**: Theoretical prediction of R_a vs. numerical results.



```
./plots/homogeneous_identical_binary_input_compos.pdf
```

Figure 11: Homogeneous Identical Binary Input. **A**: Population average of inter-neuron cross correlation. **B**: Prediction error of recurrent input variance. **C**: Theoretical prediction of R_a vs. numerical results.



`./plots/heterogeneous_identical_binary_input_compos.pdf`

Figure 12: Heterogeneous Identical Binary Input. **A**: Population average of inter-neuron cross correlation. **B**: Prediction error of recurrent input variance. **C**: Theoretical prediction of R_a vs. numerical results.