

Matrix Measure Flows: A Novel Approach to Stable Plasticity in Neural Networks

Leo Kozachkov¹, Jean-Jacques Slotine^{1,2}

¹Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology

²Department of Mechanical Engineering, Massachusetts Institute of Technology
{leokoz8, jjs}@mit.edu

Abstract—This letter introduces the notion of a *matrix measure flow* as a tool for analyzing the stability of neural networks with time-varying weights. Given a matrix flow – for example, one induced by gradient-based adaptation – the matrix measure flow tracks the progression of an associated matrix measure (or logarithmic norm). We show that for certain matrix flows of interest in computational neuroscience and machine learning, the associated matrix measure flow obeys a simple inequality. In the context of neuroscience, synapses – the connections between neurons in the brain – are constantly being updated. This plasticity subserves many important functions, such as memory consolidation and fast parameter adaptation towards real-time control objectives. However, synaptic plasticity poses a challenge for stability analyses of recurrent neural networks, which typically assume fixed synapses. Matrix measure flows allow the stability and contraction properties of recurrent neural networks with plastic synapses to be systematically analyzed. This in turn can be used to establish the robustness properties of neural dynamics, including those associated with problems in optimization and control. We consider examples of synapses subject to Hebbian and/or Anti-Hebbian plasticity, as well as covariance-based and gradient-based rules.

I. INTRODUCTION

Biological synapses are always in flux. Changes to synaptic strength can occur on timescales between hundreds of milliseconds to hours and days [1]. These changes underlie myriad important functions, such as long-term learning and fast biological adaptive control. As pointed out by [2], this poses a challenge for the brain. Synaptic plasticity inside a recurrent neural network can lead to uncontrolled instabilities. This can readily be seen for correlation-based learning rules such as Hebbian learning. If two neurons have correlated activity, the synaptic weight between them increases. Since the weight increased, their activity will correlate more, thus increasing the weight again, and so on. This is a positive feedback loop that will lead to exponential “explosions” of activity if left unchecked [1]. Since this explosion does not appear to happen in the brain, it is of interest to ask “what synaptic plasticity rules preserve overall stability”? Of course, this stability problem is not limited to biological brains. It applies broadly to machines with changing parameters.

a) Contraction Analysis: There has been much recent work in the computational neuroscience, machine learning, and control theory communities analyzing the stability of

recurrent neural network with *fixed* weights [3], [4], [5], [6], [7]. In particular, authors have been interested in restricting the weights in a recurrent network so that it is provably *contracting* [8]. Any pair of trajectories from a contracting dynamical system will converge towards one another exponentially, regardless of initial condition. Contraction implies many desirable properties for neural networks and Neural ODEs, such as exponential convergence to a unique fixed point (for autonomous systems) [9], robustness [6], and the ability to be combined with other contracting system in ways that preserve contraction [4]. In this letter we extend these analyses to models with changing parameters, which is the more biologically realistic case.

b) Neural Network: We consider an n -dimensional neural network (or Neural-ODE). The interactions between neurons in this network are governed by a single weight matrix W , through the following standard continuous-time equations

$$\tau \dot{x} = -x + W\phi(x) + u(t) \quad (1)$$

where $\tau \geq 0$ is the timescale of the neural network, ϕ is the nonlinearity, and $u(t)$ is some external input. Most applications of contraction analysis to (1) have focused on the case where the weight matrix W does not change. In this case, one can find simple restrictions on W that guarantee global contraction of (1). However, an organism’s ability to adapt to its environment crucial depends on W changing. The ways in which the weight matrix changes is typically a function of the neural states, as well as the current weight matrix. We focus on weight updates which can be written in the form

$$\dot{W} = -\gamma W + G(W, x, t) \quad (2)$$

where $\gamma \geq 0$ is a leak coefficient and G is some nonlinear matrix function. Biologically, the leak term $-\gamma W$ serves the purpose of ensuring that synaptic weights do not “blow up” [10]. It also emerges naturally in diverse settings, such as the minimization of Tikhonov regularized objective functions, as well as natural gradient descent in deep linear neural networks [11]. The term G captures interactions between the synaptic weights, neurons, as well as other external variables (e.g glia [12]). For example, a simple correlational Hebbian rule [13], [1] can be written as the outer product of firing

rate vector with itself

$$G_{Hebb} = \phi(x)\phi(x)^T$$

Taken together, the equations (1) and (2) represent a closed-loop dynamical system consisting of two interacting components: a neural component and a synaptic components. In this note, we use contraction analysis to derive sufficient conditions on the learning rule G which ensure that the combined systems (1) and (2) will remain stable.

c) *Timescale Separation*: We are interested in the case where the time constant of the neural network is small $\tau \ll 1$, or equivalently, when the dynamics of the synaptic weights are slower than those of the neurons. In this case, contraction of the overall system can be understood in the sense of singular perturbation theory, as explained in [14] and [15]. Intuitively, because we assume a separation of timescales, from the perspective of the neural dynamics (1) the synaptic weight matrix is constant. Conversely, from the perspective of the synaptic dynamics (2), the neural dynamics converge instantaneously. To maintain this interpretation, we need to ensure that the neural dynamics are contracting for all fixed W . As we will discuss in the next section, this can be achieved by ensuring the synaptic dynamics (2) force the *matrix measure* of $W(t)$ to be sufficiently small as $t \rightarrow \infty$.

II. MATRIX MEASURES AND THEIR FLOWS

A. Matrix Measure Background

The main tool we will use in our analysis is the matrix measure [16], [17]. Matrix measures have found wide application in dynamical systems and control theory, in particular in the context of contraction analysis. A matrix measure (or logarithmic norm) maps square matrices onto $\mathbb{R}$. It is a function that is induced by a corresponding matrix norm $\|\cdot\|_i$, in the following way

$$\mu_i[W] := \lim_{h \rightarrow 0^+} \frac{\|I + hW\|_i - 1}{h} \quad (3)$$

Two common matrix measures are μ_2 and μ_1 , the matrix measures associated with the matrix 2-norm and 1-norm, respectively

$$\begin{aligned} \mu_2[W] &= \lambda_{\max} \left[\frac{W + W^T}{2} \right] \\ \mu_1[W] &= \max_j \left[W_{jj} + \sum_{i \neq j} |W_{ij}| \right] \end{aligned}$$

where $\lambda_{\max}[\cdot]$ denotes the largest eigenvalue. More details can be found in [17] or [18, Section 2.2.2]. Two important properties of matrix measures $\mu[\cdot]$ are subadditivity and positive homogeneity. That is, for any two real matrices $A, B \in \mathbb{R}^{n \times n}$, we have

$$\begin{aligned} \mu[A + B] &\leq \mu[A] + \mu[B], & (\text{subadditivity}) \\ \mu[cA] &= c\mu[A] \text{ for all } c \geq 0, & (\text{positive homogeneity}) \end{aligned}$$

The reason matrix measures are important in contraction analysis is the following: if there exists a matrix measure of a system's Jacobian which is globally negative, then the system

is globally contracting. For recurrent neural networks with fixed parameters, it is usually enough to restrict the matrix measure to ensure global contraction. These restrictions are usually upper bounds on the matrix measure, of the form [19], [5], [4]

$$\mu[W] \leq k \text{ with } k > 0 \quad (4)$$

The number k is typically related to the nonlinearity ϕ used in (1). A common choice is $k = g^{-1}$ where $0 < \phi' \leq g$. For example, using this choice of k it was shown by [4] that if W is symmetric and $\mu_2[W] < g^{-1}$, then the RNN (1) is globally contracting. For non-symmetric W , it was shown in [20] that $\mu_2[W] < g^{-1}$ implies global asymptotic stability of a unique fixed point, provided that the inputs into the RNN are constant. Similarly, it was shown by [5] that $\mu_1[W] < g^{-1}$ is a sufficient condition for global contractivity of (1).

a) Eigenvalues, Matrix Measures, and Matrix Norms:

The stability of linear systems is fully characterized by the eigenvalues of the system matrix [21]. Because nonlinear systems are sometimes well-approximated by linear systems around fixed points, eigenvalues have played a large role in the stability analysis of RNNs [22], [23]. However, these analyses can typically only provide local stability results. Matrix norms and matrix measures have proven to be more useful in providing global stability results for the neural network (1). This is somewhat unsurprising, in view of the ordering between eigenvalues, matrix measures, and matrix norms. Note that for a given matrix norm $\|\cdot\|_i$ and matrix measure μ_i , one has the following chain of inequalities related to the largest real part of the eigenvalues

$$\lambda_{\max}[W] \leq \mu_i[W] \leq \|W\|_i$$

For nonlinear systems, eigenvalues tend to give liberal stability bounds, but these are only local. Matrix norms tend to give global bounds, but are often too conservative. Matrix measures span the Goldilocks zone in between these two, while still yielding global results.

B. Matrix Measure Flows

As explained above, we are interested in the situation when W evolves with time according to (2). In this setting, we ask: what happens to the *matrix measure* of W when it evolves according to (2)? The dynamics (2) induce an unknown *matrix measure flow*

$$\frac{d^+ \mu[W]}{dt} = ?$$

where $\frac{d^+}{dt}$ denotes the Dini derivative [24] (or upper right-hand derivative) at time t . The reason we consider the Dini derivative—as opposed to an ordinary derivative—is because the matrix measure may not always be differentiable (consider the μ_1 matrix measure above). It is, however, always continuous with respect to the matrix elements, which is all we need. In the following Theorem, we show that this matrix measure flow is related in a simple way to the evolution of W itself.

Theorem 1: Let $\mu[\cdot] : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}$ denote a matrix measure. Assume that $W \in \mathbb{R}^{n \times n}$ evolves according to the matrix differential equation (2). Then the following scalar differential inequality holds

$$\frac{d^+ \mu[W]}{dt} \leq -\gamma \mu[W] + \mu[G(W, x, t)] \quad (5)$$

Proof: To avoid clutter, we will write $G(W_t, x_t, t)$ as G_t . Consider the first-order Euler discretization of the matrix differential equation (2), with step-size h such that $h < \frac{1}{\gamma}$,

$$W_{t+h} = W_t - h\gamma W_t + hG_t = (1 - h\gamma)W_t + hG_t$$

Using the sub-additivity property of matrix measures, we have

$$\mu[W_{t+h}] = \mu[(1 - h\gamma)W_t + hG_t] \leq (1 - h\gamma)\mu[W_t] + h\mu[G_t]$$

where the last step uses the positive homogeneity of matrix measures to ‘pull out’ the $1 - h\gamma$ and h terms. Subtracting $\mu[W_t]$ from both sides of this inequality and dividing both sides by h , we see that

$$\frac{\mu[W_{t+h}] - \mu[W_t]}{h} \leq -\gamma \mu[W_t] + \mu[G_t]$$

Since the matrix measure $\mu[W_t]$ may not always be differentiable with respect to time, we consider the right-hand supremum limit (i.e., the Dini derivative) to arrive at the stated differential inequality

$$\frac{d^+ \mu[W]}{dt} = \limsup_{h \rightarrow 0^+} \frac{\mu[W_{t+h}] - \mu[W_t]}{h} \leq -\gamma \mu[W] + \mu[G(W, x, t)]$$

Remark 1: Theorem 1 holds even when the leak rate γ depends on W and t , so long as $\gamma(W, t) \geq 0$.

Remark 2: Theorem 1 is applicable for *any* function $\mu[\cdot] : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}$ such that $\mu[\cdot]$ is subadditive and positively homogeneous. This includes, for example, any matrix norm.

Corollary 1: Assume $\gamma > 0$ and $\mu[G(W, x, t)] \leq D$, for some $D \in \mathbb{R}$. Then equation (5) implies an upper-bound on $\mu[W(t)]$

$$\mu[W(t)] \leq \mu[W(0)]e^{-\gamma t} + \frac{D}{\gamma} \quad (6)$$

This is a consequence of the Grönwall Comparison Lemma (see Exercise E2.1 in [25] for an explicit proof). From (6) we can see that the inequality $\mu[W(t)] \leq k$ will be satisfied in finite time if the following strict inequality between D , k and γ is satisfied

$$\frac{D}{k} < \gamma \quad (7)$$

This strict inequality is necessary to ensure that $\mu[W(t)]$ eventually drops below k , instead of merely approaching it from above.

III. APPLICATIONS

Theorem 1 has immediate application to the problem of determining contraction in neural networks with synaptic plasticity.

A. Anti-Hebbian Plasticity

Consider the simple anti-Hebbian plasticity rule analyzed in [26] and [27], with

$$\dot{W} = -\gamma W - \phi(x)\phi(x)^T$$

Using the notation of (2), we can identify

$$G(W, x, t) = -\phi(x)\phi(x)^T$$

Since $G(W, t)$ is a negative semi-definite matrix, we have that $\mu_2[G] \leq 0$. This implies that D in Corollary 1 may be taken to be $D = 0$. Substituting $D = 0$ into equation (6), we see that $\mu_2[W(t)]$ decays exponentially to zero

$$\mu_2[W(t)] \leq \mu_2[W(0)]e^{-\gamma t}$$

Note that the above exponential decay does *not* imply that W itself decays to zero. It simply implies that largest eigenvalue of the symmetric part of W decays towards zero. Since G is symmetric, we can also deduce that if $W(t)$ is initialized as a symmetric matrix, then $W(t)$ will be symmetric for all time. If W is initially asymmetric, then it will converge exponentially with rate γ towards a symmetric matrix. This follows from decomposing W into a symmetric and skew-symmetric matrix, and then noting that the skew-symmetric part decays exponentially with rate γ .

B. Hebbian/Anti-Hebbian Synaptic Plasticity

The brain has both Hebbian and Anti-Hebbian synaptic plasticity [28], [10], [29]. Theorem 1 gives simple sufficient conditions for what combinations of Hebbian and Anti-Hebbian plasticity lead to contracting weight matrices. In particular, consider

$$\dot{W} = -K \odot \phi(x)\phi(x)^T - \gamma W$$

where $K \in \mathbb{R}^{n \times n}$ is an arbitrary positive semi-definite matrix, and $\odot$ denotes element-wise matrix multiplication (i.e. the Hadamard product). These dynamics were also initially analyzed in [26], but there the signs of the entries of K were constrained to be positive. Here we do not constrain the signs of the individual matrix entries of K , so this plasticity rule is both Hebbian and/or Anti-Hebbian. In this case, we have

$$G(W, t) = -K \odot \phi(x)\phi(x)^T$$

Since $-K$ is negative semi-definite by assumption and $\phi(x)\phi(x)^T$ is positive semi-definite by construction, the Schur Product Theorem [30] tells us that their elementwise product will be negative semi-definite

$$\mu_2[-K \odot \phi(x)\phi(x)^T] \leq 0$$

This again implies that $D = 0$. As in the previous section, this in turn implies that the largest eigenvalue of the symmetric part of W will decay to zero exponentially. Additionally, the symmetry of K implies that W will remain symmetric if initialized as a symmetric matrix, or will converge exponentially with rate γ towards a symmetric matrix if initialized as an asymmetric matrix.

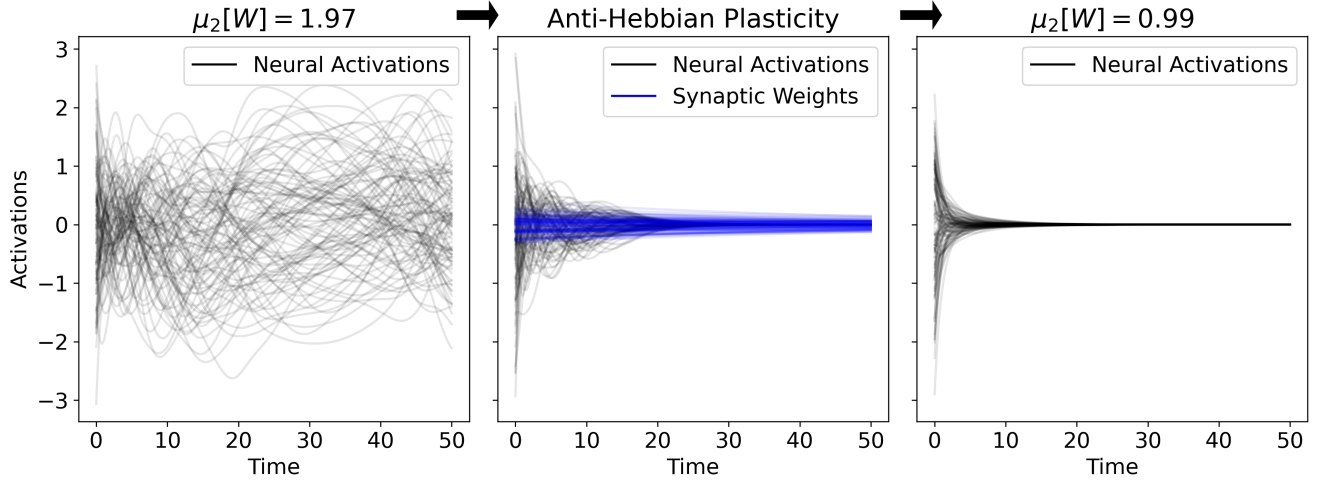


Fig. 1. Left) Chaotic neural dynamics for the RNN described by (1), associated with a large matrix measure for the synaptic weight matrix. Each trace the time-dependent activity of a neuron, as the network evolves from a randomly chosen initial condition. Center) Anti-Hebbian plasticity induces a stabilizing matrix measure flow. This flow forces the matrix measure of the weight matrix to be less than unity, leading to stable neural dynamics. See subsection III-B for details. Right) After the synaptic dynamics are run for a sufficiently long (yet finite) time, the weight matrix is such that the associated neural dynamics are stable. In this case the neural dynamics converge exponentially towards a unique, global attractor.

Note that we may also include a matrix bias term $B = B^T$ in the synaptic dynamics

$$G(W, t) = -K \odot \phi(x)\phi(x)^T + B$$

From the sub-additivity of the matrix measure, the inclusion of B increases the disturbance term D by at worst $\mu[D]$. See Figure 1 for a numerical example, with $n = 100$, $\phi(x) = \tanh(x)$, $K_{ij} = 1$ and $B_{ij} = 0.9/n$.

C. Dong & Hopfield Synaptic Plasticity

The previous two examples relied on the positive semi-definiteness of the system matrix K , which in turn implies that $D \leq 0$. In this example we consider a case where $D \geq 0$. Consider the following Hebbian synaptic dynamics, first introduced and analyzed by [13]

$$\dot{W} = \nu \phi(x)\phi(x)^T - \gamma W$$

where $\nu > 0$ is a learning rate. Consider for example the bounded activation $\phi(\cdot) = \tanh(x)$. Applying the matrix measure associated with the 2-norm to

$$G(W, t) = \nu \phi(x)\phi(x)^T$$

we can see that $D \leq \nu \phi(x)^T \phi(x) \leq \nu n$. This follows from the observation that the only non-zero eigenvalue of G is $\nu \phi(x)^T \phi(x)$, and that ϕ is bounded between negative one and one. From this we can conclude that if

$$\frac{\nu n}{k} < \gamma$$

the weight matrix will satisfy the inequality $\mu_2[W(t)] \leq k$ in finite time.

D. Covariance Rule

Consider a covariance based learning rule [31], [10], with a leak term

$$\dot{W} = \nu [\phi(x) - \langle \phi \rangle] [\phi(x) - \langle \phi \rangle]^T - \gamma W$$

where $\nu \geq 0$ is a learning rate and $\langle \phi \rangle$ denotes a running average over the firing rate

$$\langle \phi \rangle_i = \int_{t-\Delta}^t \phi(x(s))_i ds$$

with some window size $\Delta > 0$. As in the section above, we can identity $D = \nu \|\phi(x) - \langle \phi \rangle\|^2$ using the matrix measure associated with the 2-norm. If we assume that the firing rate does not deviate more than σ^2 from its running average, then the strict inequality

$$\frac{\nu \sigma^2}{k} < \gamma$$

implies that the weight matrix will satisfy the inequality $\mu_2[W(t)] \leq k$ in finite time.

E. Presynaptic Plasticity

In some models [32], [33], synaptic plasticity only depends on the activity of the presynaptic neuron

$$\dot{W}_{ij} = -\gamma W_{ij} + b_i \phi(x_j)$$

where b_i is some gain parameter. If we assume that the firing rate is bounded, such that $|\phi(x_j)| \leq \phi_{max}$, and the gain term is bounded such that $|b_i| \leq b_{max}$ then application of the matrix measure associated with the 1-norm yields the strict inequality

$$\frac{n b_{max} \phi_{max}}{k} < \gamma$$

F. Gradient Descent

Many loss functions used in machine learning include a regularization term to control the model complexity. The total loss minimized during training is the sum of a “task” loss term, which measures how poorly the model is doing, and a regularization term

$$\mathcal{L} = \mathcal{L}_{task} + \frac{\gamma}{2} \mathcal{L}_{reg}$$

where $\mathcal{L}_{task}$ denotes the loss purely due to performance error, and $\mathcal{L}_{reg}$ denotes the regularization term. A common choice of regularization is the squared sum-of-squares of all the weights (i.e the squared Frobenius norm). For square matrices this can be written as

$$\mathcal{L}_{reg} = \text{Tr}(W^T W) = \|W\|_{fro}^2$$

Gradient flow on this loss function yields the following familiar expression for $\dot{W}$

$$\begin{aligned} \dot{W} &= -\nabla_W \mathcal{L} = -\nabla_W \mathcal{L}_{task} - \frac{\gamma}{2} \nabla_W \mathcal{L}_{reg} \\ &= -\nabla_W \mathcal{L}_{task} - \gamma W \end{aligned}$$

If we assume that the loss function is Lipschitz with respect to the parameters W , with Lipschitz constant L , then it is straightforward to show that making γ large enough will produce a weight matrix with a matrix measure less than k . Using the fact that the matrix measure is always upper-bounded by its associated matrix norm, we have

$$\mu_i[G] = \mu_i[-\nabla_W \mathcal{L}_{task}] \leq \|-\nabla_W \mathcal{L}_{task}\|_i \leq L$$

This shows that we may take D in Corollary (1) to be equal to $D = L$. To satisfy the inequality for k , we simply need to ensure that

$$\frac{L}{k} < \gamma$$

IV. CONCLUSION

Understanding how the brain balances stability and plasticity is an old and important problem in computational neuroscience [2]. We introduce the notion of a matrix measure flow and show that it is useful for approaching this problem. While our analysis has been focused on recurrent neural networks, the basic idea of a matrix measure flow can be applied to any system with “plastic” weights (for example, biochemical reactions characterized by changing rate parameters [34], [35]). Similarly, our results may be useful in the context of adaptive control and prediction, where certain adaptive laws may be thought of as gradient flow [36]. In this case, the assumption of timescale separation may be relaxed.

ACKNOWLEDGMENTS

We would like to thank Francesco Bullo for helpful suggestions and comments. L.K. is supported by a K. Lisa Yang Integrative and Computational Neuroscience (ICoN) Postdoctoral Fellowship.

REFERENCES

- [1] P. Dayan and L. F. Abbott, *Theoretical neuroscience: computational and mathematical modeling of neural systems*. MIT press, 2005.
- [2] F. Zenke, W. Gerstner, and S. Ganguli, “The temporal paradox of hebbian learning and homeostatic plasticity,” *Current opinion in neurobiology*, vol. 43, pp. 166–176, 2017.
- [3] M. Revay and I. Manchester, “Contracting implicit recurrent neural networks: Stable models with improved trainability,” in *Learning for Dynamics and Control*. PMLR, 2020, pp. 393–403.
- [4] L. Kozachkov, M. M. Ennis, and J.-J. Slotine, “Rnns of rnns: Recursive construction of stable assemblies of recurrent neural networks,” in *Advances in Neural Information Processing Systems*, 2021.
- [5] A. Davydov, A. V. Proskurnikov, and F. Bullo, “Non-euclidean contractivity of recurrent neural networks,” in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 1527–1534.
- [6] S. Jafarpour, A. Davydov, A. Proskurnikov, and F. Bullo, “Robust implicit networks via non-euclidean contractions,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 9857–9868, 2021.
- [7] I. R. Manchester, M. Revay, and R. Wang, “Contraction-based methods for stable identification and robust machine learning: a tutorial,” in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 2955–2962.
- [8] W. Lohmiller and J.-J. E. Slotine, “On contraction analysis for nonlinear systems,” *Automatica*, vol. 34, no. 6, pp. 683–696, 1998.
- [9] J.-J. E. Slotine, “Modular stability tools for distributed computation and control,” *International Journal of Adaptive Control and Signal Processing*, vol. 17, no. 6, pp. 397–416, 2003.
- [10] W. Gerstner and W. M. Kistler, “Mathematical formulations of hebbian learning,” *Biological cybernetics*, vol. 87, no. 5, pp. 404–415, 2002.
- [11] A. Bernacchia, M. Lengyel, and G. Hennequin, “Exact natural gradient in deep linear networks and its application to the nonlinear case,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [12] A. Araque, G. Carmignoto, P. G. Haydon, S. H. Oliet, R. Robitaille, and A. Volterra, “Gliotransmitters travel in time and space,” *Neuron*, vol. 81, no. 4, pp. 728–739, 2014.
- [13] D. W. Dong and J. J. Hopfield, “Dynamic properties of neural networks with adapting synapses,” *Network: Computation in Neural Systems*, vol. 3, no. 3, p. 267, 1992.
- [14] D. Del Vecchio and J.-J. E. Slotine, “A contraction theory approach to singularly perturbed systems,” *IEEE Transactions on Automatic Control*, vol. 58, no. 3, pp. 752–757, 2012.
- [15] H. D. Nguyen, T. L. Vu, K. Turitsyn, and J.-J. Slotine, “Contraction and robustness of continuous time primal-dual dynamics,” *IEEE control systems letters*, vol. 2, no. 4, pp. 755–760, 2018.
- [16] G. Dahlquist, “Stability and error bounds in the numerical integration of ordinary differential equations,” Ph.D. dissertation, Almqvist & Wiksell, 1958.
- [17] C. Desoer and H. Haneda, “The measure of a matrix as a tool to analyze computer algorithms for circuit analysis,” *IEEE Transactions on Circuit Theory*, vol. 19, no. 5, pp. 480–486, 1972.
- [18] M. Vidyasagar, *Nonlinear systems analysis*. SIAM, 2002.
- [19] Y. Fang and T. G. Kincaid, “Stability analysis of dynamical neural networks,” *IEEE Transactions on Neural Networks*, vol. 7, no. 4, pp. 996–1006, 1996.
- [20] K. Matsuoka, “Stability conditions for nonlinear continuous neural networks with asymmetric connection weights,” *Neural networks*, vol. 5, no. 3, pp. 495–500, 1992.
- [21] D. G. Luenberger, *Introduction to dynamic systems: theory, models, and applications*. Wiley New York, 1979, vol. 1.
- [22] H. Jaeger, “The “echo state” approach to analysing and training recurrent neural networks—with an erratum note,” *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, vol. 148, no. 34, p. 13, 2001.
- [23] G. Christodoulou and T. Vogels, “The eigenvalue value (in neuroscience),” *OSF Preprints*, 2022.
- [24] H. K. Khalil, *Nonlinear control*. Pearson New York, 2015, vol. 406.
- [25] F. Bullo, *Contraction Theory for Dynamical Systems*, 1.0 ed. Kindle Direct Publishing, 2022. [Online]. Available: <http://motion.me.ucsb.edu/book-ctds>
- [26] L. Kozachkov, M. Lundqvist, J.-J. Slotine, and E. K. Miller, “Achieving stable dynamics in neural circuits,” *PLoS computational biology*, vol. 16, no. 8, p. e1007659, 2020.
- [27] V. Centorrino, A. Gokhale, A. Davydov, G. Russo, and F. Bullo, “Euclidean contractivity of neural networks with symmetric weights,” *IEEE Control Systems Letters*, 2023.

- [28] C. C. Bell, A. Caputi, K. Grant, and J. Serrier, "Storage of a sensory pattern by anti-hebbian synaptic plasticity in an electric fish," *Proceedings of the National Academy of Sciences*, vol. 90, no. 10, pp. 4650–4654, 1993.
- [29] C. Pehlevan, T. Hu, and D. B. Chklovskii, "A hebbian/anti-hebbian neural network for linear subspace learning: A derivation from multi-dimensional scaling of streaming data," *Neural computation*, vol. 27, no. 7, pp. 1461–1495, 2015.
- [30] F. Zhang, *The Schur complement and its applications*. Springer Science & Business Media, 2006, vol. 4.
- [31] T. J. Sejnowski and G. Tesauro, "The hebb rule for synaptic plasticity: algorithms and implementations," in *Neural models of plasticity*. Elsevier, 1989, pp. 94–103.
- [32] G. Mongillo, O. Barak, and M. Tsodyks, "Synaptic theory of working memory," *Science*, vol. 319, no. 5869, pp. 1543–1546, 2008.
- [33] N. Y. Masse, G. R. Yang, H. F. Song, X.-J. Wang, and D. J. Freedman, "Circuit mechanisms for the maintenance and manipulation of information in working memory," *Nature neuroscience*, vol. 22, no. 7, pp. 1159–1167, 2019.
- [34] G. Russo, M. Di Bernardo, and E. D. Sontag, "Global entrainment of transcriptional systems to periodic inputs," *PLoS computational biology*, vol. 6, no. 4, p. e1000739, 2010.
- [35] A. Raveh, M. Margaliot, E. D. Sontag, and T. Tuller, "A model for competition for ribosomes in the cell," *Journal of The Royal Society Interface*, vol. 13, no. 116, p. 20151062, 2016.
- [36] N. M. Boffi, S. Tu, and J.-J. E. Slotine, "Random features for adaptive nonlinear control and prediction," *arXiv preprint arXiv:2106.03589*, 2021.