Is All Learning (Natural) Gradient Descent?

Lucas Shoji[1], Kenta Suzuki[2], Leo Kozachkov[3,4,*]

[1]Department of Physics, MIT
[2]Department of Mathematics, MIT
[3]Thomas J. Watson Research Center, IBM Research
[4]Department of Brain and Cognitive Sciences, MIT
[*]Corresponding Author
leokoz8@[ibm.com,mit.edu]

Abstract

This paper shows that a wide class of effective learning rules—those that improve a scalar performance measure over a given time window—can be rewritten as natural gradient descent with respect to a suitably defined loss function and metric. Specifically, we show that parameter updates within this class of learning rules can be expressed as the product of a symmetric positive definite matrix (i.e., a metric) and the negative gradient of a loss function.

We also demonstrate that these metrics have a canonical form and identify several optimal ones, including the metric that achieves the minimum possible condition number. The proofs of the main results are straightforward, relying only on elementary linear algebra and calculus, and are applicable to continuous-time, discrete-time, stochastic, and higher-order learning rules, as well as loss functions that explicitly depend on time.

1 Introduction

Finding biologically plausible learning rules for ecologically-relevant tasks is a major goal in neuroscience [1, 2], just as identifying effective training rules for large-scale neural networks is in machine learning and artificial intelligence. This paper does not offer either. Instead, we demonstrate that *if* such rules are found, then under fairly mild assumptions (i.e., continuous or small updates), they can be written in a very specific form: the product of a symmetric, positive definite matrix and the negative gradient of a loss function.

It is well-known that if a learning rule updates parameters by following the negative gradient of a loss function, the loss decreases along the parameter trajectories [3, 4]. However, many learning rules do not fit this "pure" gradient descent form. Indeed, there are compelling reasons to believe that the brain's learning rules *cannot* be expressed as pure gradient descent

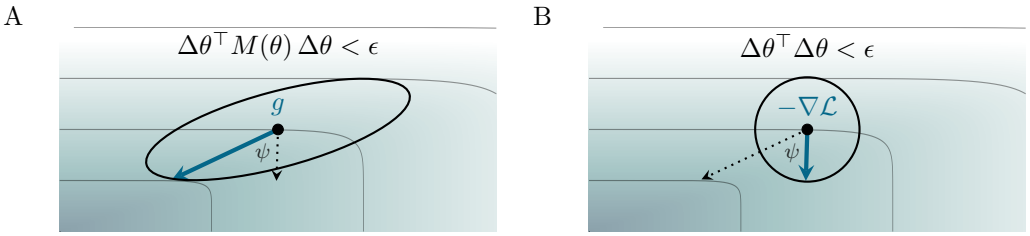

Figure 1: A) Contour lines of a loss function (darker colors = lower loss). Parameters update in the direction of g. If this update decreases the loss, and if the step-size is small, g is equivalent to steepest descent with a non-Euclidean metric, $M(\theta)$. In this case, the angle ψ between g and the negative gradient is acute. Ellipse: ϵ-ball in this metric. B) Steepest descent with the Euclidean metric. Circle: ϵ-ball in this metric.

[1, 5, 6]. Fortunately, there are many ways to decrease a loss function beyond traditional gradient descent. One notable class of algorithms, which we focus on in this paper, is natural gradient descent [7].

In natural gradient algorithms, parameter updates are written as the product of a symmetric positive definite matrix and the negative gradient. If a learning rule can be expressed in this form, it is considered "effective" because it guarantees improvement of a scalar performance measure over time (assuming small step sizes). Given the flexibility of choosing the positive definite matrix, one can ask the converse question: if a learning rule is effective, can it be written as natural gradient descent? We show that for a wide class of effective learning rules this is indeed the case. For example, our results hold for *all* effective continuous-time learning rules. This leads us to conjecture that *any* sequence of updates which improves a scalar measure of performance can be written in natural gradient form.

Formal Setting We consider a set of D real numbers $\theta \in \mathbb{R}^D$ which parameterize the function of a system. In the case of biology, these numbers can represent biophysical variables such as synaptic diffusion constants or receptor densities [2]. In the case of artificial neural networks, these numbers can represent synaptic weights between units. We analyze two common methods for updating θ towards the goal of improving performance on a task (or set of tasks): continuous-time evolution and discrete-time updates. In the former, θ evolves continuously according to a flow

$$\frac{d\theta}{dt} = g(\theta, t) \tag{1}$$

where $g(\theta, t)$ is a potentially nonlinear, time-dependent function. At this stage, we impose no restrictions on this function (e.g., smoothness). In discrete-time updates, changes to θ occur at discrete time intervals

$$\theta_{t+1} = \theta_t + \eta\, g(\theta_t, t) \tag{2}$$

where $\eta > 0$ is a learning rate parameter. This setting is general enough to capture supervised learning, self-supervised learning, as well as in-context learning (where t may be identified with layers in a neural network). Also note that (1) and (2) include techniques which rely on defining higher-order derivatives of θ, such as accelerated gradient methods [8]. In this case, one can arrive back at the form of (1) and (2) by expanding the state space [1].

Effective Learning Rules Do Not Require Monotonic Improvement We assume that each θ can be associated with a system which performs some task. For example, suppose θ contains the weights of a neural network after training. This neural network can then be evaluated based on its performance on some task. We define an effective learning rule as one which leads to the improvement of a scalar performance measure over some time window. We will use the loss $\mathcal{L}$ to measure performance, so that "improvement" means the loss decreases after time m has elapsed

$$\mathcal{L}(t+m) < \mathcal{L}(t). \tag{3}$$

Note that this definition does not require monotonic improvement in the performance measure. In particular, (3) allows for temporary setbacks, i.e., $d\mathcal{L}/dt > 0$, so long as the setbacks do not outweigh the progress on average. This includes, for example, learning rules that take "one step backwards, two step forwards". Note also that although the loss does not decrease monotonically along trajectories of (1), the average loss $\mathcal{L}_{\text{avg}}$ does, because

$$\mathcal{L}_{\text{avg}} \coloneqq \frac{1}{m}\int_t^{t+m} \mathcal{L}(s)\, ds \qquad \Longrightarrow \qquad \dot{\mathcal{L}}_{\text{avg}} = \frac{\mathcal{L}(t+m) - \mathcal{L}(t)}{m} < 0$$

where the inequality was obtained by using assumption (3). The same argument can be applied to discrete-time updates. In this case, the average loss continually improves, because

$$\mathcal{L}_{\text{avg}}(t) = \frac{1}{m}\sum_{\tau=t}^{t+m-1} \mathcal{L}(\tau) \qquad \Longrightarrow \qquad \mathcal{L}_{\text{avg}}(t+1) - \mathcal{L}_{\text{avg}}(t) = \frac{\mathcal{L}(t+m) - \mathcal{L}(t)}{m} < 0.$$

[1] E.g., for second-order methods, define the extended state space $\begin{bmatrix} v & \theta \end{bmatrix}$, where $v \coloneqq \dot{\theta}$.

Therefore, for the remainder of the paper we will assume without loss of generality that the loss function $\mathcal{L}$ *does* monotonically decrease. Also note that while the average loss is a particularly convenient measure of asymptotic improvement, we can in fact consider much more general measures that guarantee asymptotic improvement of a performance measure without continual improvement–for example, by considering a sequence of Lyapunov functions as done by Ahmadi and Parrilo [9]. Finally, while we only consider differentiable loss functions in this paper, analogous results hold for non-differentiable losses using suitable replacements for the gradient of the loss [10].

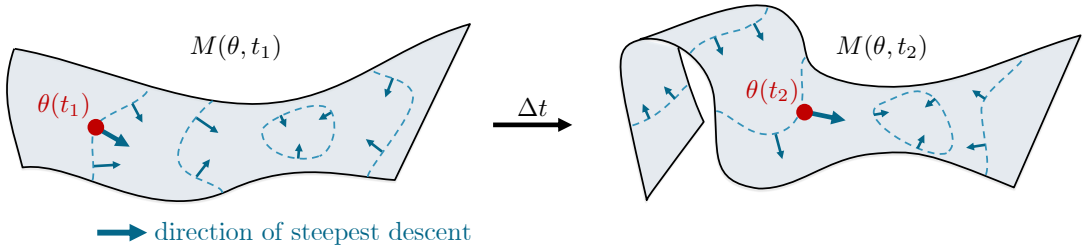

Figure 2: Natural gradient descent minimizes a loss function (dashed contours) by evolving the parameters θ in the direction of steepest descent in a non-Euclidean space. This space, a D-dimensional manifold with metric $M(\theta, t)$, is visualized as a surface embedded in a higher dimensional Euclidean space. We demonstrate that a wide class of learning rules that decreases the loss function (not necessarily monotonically) fits this framework. In this context, the dynamics of both θ and M are determined by the learning rule and the loss function.

(Natural) Gradient Descent Gradient descent is a prototypical algorithm for decreasing a loss function. However it is by no means the only algorithm which does so. An important generalization of gradient descent is *natural* gradient descent [7]

$$\dot{\theta} = -M^{-1}(\theta, t)\ \nabla_\theta \mathcal{L} \tag{4}$$

where $M(\theta, t)$ is some symmetric positive definite matrix[2]. To see that natural gradient descent indeed decreases the loss $\mathcal{L}$ in continuous-time, suppose that θ is not at a stationary point, i.e., $\|\nabla_\theta \mathcal{L}\| > 0$. Then,

$$\dot{\mathcal{L}} = \nabla_\theta \mathcal{L}^\top\ \dot{\theta} = -\nabla_\theta \mathcal{L}^\top M^{-1}(\theta, t)\ \nabla_\theta \mathcal{L} \leq -\frac{\|\nabla_\theta \mathcal{L}(s)\|^2}{\lambda_{\max}(M)} < 0, \tag{5}$$

where $\lambda_{\max}(M) > 0$ denotes the largest eigenvalue of M. The first equality follows from the chain rule, the second equality follows from substituting in (4), and the inequality is obtained by using the Rayleigh quotient [11]. The above conclusion also holds in discrete-time, for sufficiently small learning rate η.

There are two interesting connections between natural gradient descent and gradient descent. The first is that in the special case when $M = I$, natural gradient descent reduces to gradient descent. The second connection is that both gradient descent and natural gradient descent perform steepest descent: the negative gradient is the direction of steepest descent in Euclidean space, whereas the negative natural gradient denotes the direction of steepest descent in some non-Euclidean space. In particular, in a space where unit lengths at point θ satisfy

$$a^\top M(\theta, t)\ a = 1.$$

add figure reference Natural gradients underlie many techniques in machine learning and optimization [12–18], control theory [17, 19–21], and, more recently, have enjoyed renewed interest in neuroscience [1, 5, 22].

[2]Technically, this is natural gradient *flow*. We will use the term descent to refer to both continuous and discrete updates.

2 Main Results

2.1 Continuous-Time Learning Rules

To streamline the notation, we define y as the negative gradient of $\mathcal{L}$ and the update vector $g(\theta, t)$, defined in (1), as g. This allows us to express the monotonic decrease of the loss function more concisely as $y^\top g > 0$. Our goal is to find a symmetric positive definite matrix M that maps g to y, which ensures that g can be written in the natural gradient form

$$Mg = y \qquad \Longleftrightarrow \qquad g = -M^{-1}\,\nabla_\theta \mathcal{L}.$$

Towards this goal, consider the matrix

$$M = \frac{1}{y^\top g} yy^\top + \sum_{i=1}^{D-1} u_i u_i^\top. \tag{6}$$

Here, the vectors u_i are chosen to span the subspace orthogonal to g, denoted by $g^\perp \coloneqq \{v \in \mathbb{R}^n : v^\top g = 0\}$. As desired, M maps the update vector g to the negative gradient direction y. By construction, M is symmetric and positive definite. Indeed, for any non-zero vector x we have

$$x^\top M x = \frac{1}{y^\top g}\left(x^\top y\right)^2 + \sum_{i=1}^{D-1} \left(x^\top u_i\right)^2 > 0.$$

The inequality holds because x cannot be simultaneously orthogonal to both y and all the u_i, as this would contradict the assumption that $y^\top g > 0$. Later on, for a special family of metrics, we will derive the full spectrum of M.

Canonical form of the Metric We will now show that *any* symmetric, positive definite matrix M such that $Mg = y$, with $g^\top y > 0$, is of the form given in (6).

Proof. Let M satisfy the requirements given above. Define

$$M' \coloneqq M - \frac{1}{y^\top g} yy^\top$$

which is a symmetric matrix. We claim that for any nonzero $u \in g^\perp$,

$$u^\top M' u > 0.$$

If so, since the matrix is symmetric and an "orthogonal" eigendecomposition exists, it follows that M' is of the form $\sum_{i=1}^{D-1} u_i u_i^\top$ for some basis $\{u_i\}$ of $g^\perp$, proving the canonical form. To show this, first note that

$$M'g = Mg - \frac{1}{y^\top g} yy^\top g = 0. \tag{7}$$

Now take an arbitrary nonzero $u \in g^\perp$. Consider the projection of u to $y^\perp$ along g

$$u' = u - \frac{y^\top u}{y^\top g} g,$$

which is nonzero and orthogonal to y.[3] Together with (7) we see

$$u^\top M' u = (u')^\top M' u' = (u')^\top M u' > 0,$$

concluding our proof. □

[3] This projection is well-defined by the assumption of effective learning, i.e., that $y^\top g > 0$.

One-Parameter Family of Metrics Although the matrix M in (6) is positive definite, it will be useful later to have an explicit expression for the eigenvalues of M, for example, in terms of the angle between y and g. While this is challenging for a general M, we observe that a one-parameter family of valid metrics M can be written as

$$M \;=\; \frac{1}{y^\top g} yy^\top + \alpha \sum_{i=1}^{D-1} u_i u_i^\top \;=\; \frac{1}{y^\top g} yy^\top + \alpha\left(I - \frac{gg^\top}{g^\top g}\right) \tag{8}$$

where $\alpha > 0$ can depend on y and g, and $u_i^\top u_j = \delta_{ij}$. These are exactly the matrices M which acts as the scalar α on the orthogonal complement of the span of g and y. We show in appendix A that the full spectrum of M can be derived for this family of metrics, as a function of α.

Optimal Metrics We further show in appendix A that the one-parameter family (8) contains several globally "optimal" metrics. In particular, we prove that among *all possible metrics*, not just within this one-parameter family, the metric M_{opt} which achieves the smallest condition number is given by setting $\alpha = \frac{y^\top y}{g^\top y}$ in (8). The spectrum of M_{opt} can be written in terms of the angle between y and g, which we call $\psi \in (-\frac{\pi}{2}, \frac{\pi}{2})$, as follows:

$$\begin{aligned} \lambda_{\max\,/\,\min}(M_{\text{opt}}) &= \frac{||y||}{||g||}\left[\frac{1}{\cos(\psi)} \pm |\tan(\psi)|\right] \\ \lambda_d(M_{\text{opt}}) &= \frac{||y||}{||g||}\frac{1}{\cos(\psi)} \end{aligned} \tag{9}$$

where $1 < d < D$. See Figure 3 for a plot of these curves. The condition number κ of the optimal metric M_{opt} has a particularly simple form as a function of ψ

$$\kappa(M_{\text{opt}}) = \frac{\lambda_{\max}(M_{\text{opt}})}{\lambda_{\min}(M_{\text{opt}})} = \frac{1 + |\sin(\psi)|}{1 - |\sin(\psi)|}.$$

Note that $1/\kappa(M_{\text{opt}}) \in (0, 1]$, and can be naturally viewed as a measure of similarity between g and y. We also show in appendix A that, among all possible metrics, the one with the *minimum* possible $\lambda_{\max}(M)$ is asymptotically approached as $\alpha \to 0$. It can be shown that this minimum is given by

$$\lambda_{\max}(M) > \frac{\|y\|}{\|g\|}\frac{1}{\cos(\psi)},$$

Similarly, the metric with the *maximum* possible $\lambda_{\min}(M)$ is approached asymptotically when $\alpha \to \infty$. This maximum is given by

$$\lambda_{\min}(M) < \frac{\|y\|}{\|g\|}\cos(\psi),$$

These results will be particularly useful later on, particularly when analyzing discrete-time learning rules in section 2.2.

Metric Asymptotics It is clear from (6) that the metric M will "blow-up" if the negative gradient y becomes orthogonal to the parameter update g. This is expected because, in this case, learning does not occur ($d\mathcal{L}/dt = 0$). Furthermore, in this case we would have that

$$y^\top g = g^\top M g = 0,$$

which contradicts the positive-definiteness of M. This can be confirmed by inspecting the eigenvalues of the metric M given in (9) and Figure 3. One sees that as the angle ψ between y and g approaches $\pi/2$ or $-\pi/2$, the smallest eigenvalue of the metric goes to zero, causing M to lose its positive definiteness, while the remaining eigenvalues tend to infinity.

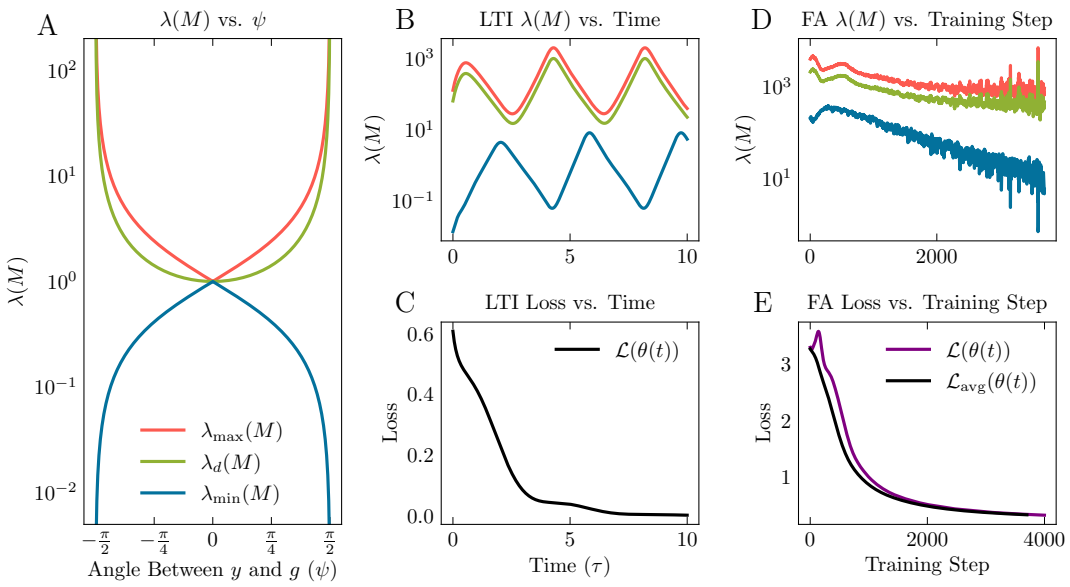

Figure 3: A) Eigenvalues of the optimal metric M_{opt} as a function of the angle ψ between vectors y and g, with the norm ratio $\|y\|/\|g\|$ fixed at unity. Refer to Eq. (9) in the main text. B) Spectrum of M_{opt} for stable linear time-invariant dynamics over time. C) Lyapunov function (loss) corresponding to the dynamics in (B), demonstrating a monotonic decrease. D) Spectrum of M_{opt} for a small multi-layer network trained with a biologically plausible learning rule (feedback alignment) to classify MNIST digits. E) Training loss of feedback alignment as a function of training steps, showing that while the instantaneous loss is not strictly monotonic, the average loss decreases over time.

Time-Varying Loss So far, we have only considered loss functions $\mathcal{L}(\theta)$ which do not depend explicitly on the time t. However, there are many cases of interest where the loss can be thought of as changing in time, for example in online convex optimization [23]. In this case, we can show that effective learning implies an *extended* parameter vector may be written as natural gradient descent on the time-varying loss $\mathcal{L}(\theta, t)$. We define this new extended vector as v, and its time-derivative as $\dot{v}$

$$v \coloneqq \begin{bmatrix} \theta & t \end{bmatrix}^\top \qquad \Longrightarrow \qquad \dot{v} = \begin{bmatrix} \dot{\theta} & 1 \end{bmatrix}^\top$$

Then, the total derivative of the time-varying loss, as θ evolves in time is given by

$$\dot{\mathcal{L}} = \frac{\partial \mathcal{L}}{\partial \theta}^\top \dot{\theta} + \frac{\partial \mathcal{L}}{\partial t} = \frac{\partial \mathcal{L}}{\partial v}^\top \dot{v} < 0$$

Thus, we may conclude that updates to the extended variable v perform natural gradient descent on the time-varying loss $\mathcal{L}$

$$\dot{v} = -M^{-1} \frac{\partial \mathcal{L}}{\partial v}$$

where M is constructed as before, with $y = -\frac{\partial \mathcal{L}}{\partial v}$ and $g = \dot{v}$.

2.2 Discrete-Time Learning Rules

Consider a discrete-time learning rule which decreases a loss function $\mathcal{L}$ at every step

$$\theta_{t+1} = \theta_t + \eta \; g(\theta_t, t) \qquad \text{and} \qquad \mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t) < 0. \tag{10}$$

In this section we will show that (10) implies that the updates g can always be written in the form of a positive definite matrix multiplied by the *discrete gradient*, which we define below. We will also show that for smooth loss functions $\mathcal{L}$ and sufficiently small η, it is possible to

construct at every time t a symmetric positive definite matrix M (in general different from the M considered above) such that

$$g(\theta_t) = -M^{-1}\ \nabla_\theta \mathcal{L}(\theta_t).$$

To prove this, we recall Taylor's Theorem [4, 24], and the definition of a Discrete Gradient [25, 26].

Theorem 1 (Taylor's Theorem)**.** *Suppose that $\mathcal{L}\colon \mathbb{R}^D \to \mathbb{R}$ is a twice continuously differentiable function, and that $p \in \mathbb{R}^D$. Then there exists some $\lambda \in (0,1)$ such that*

$$\mathcal{L}(x+p) = \mathcal{L}(x) + p^\top \nabla \mathcal{L}(x) + \frac{1}{2} p^\top\, \nabla^2 \mathcal{L}\left(x + \lambda p\right)\, p. \tag{11}$$

It is important to note that (11) is an *equality*, and not an approximation (although it can certainly be used to generate an excellent approximation of the difference between $\mathcal{L}(x+p)$ and $\mathcal{L}(x)$ when the norm of p is small and $\mathcal{L}$ is smooth).

Definition 1 (Discrete Gradient)**.** *Suppose that $\mathcal{L}\colon \mathbb{R}^D \to \mathbb{R}$ is a differentiable function, and that $p \in \mathbb{R}^D$. Then $\bar{\nabla}\mathcal{L}\colon \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}^D$ is a discrete gradient of $\mathcal{L}$ if it is continuous and*

$$\begin{cases} p^\top \bar{\nabla}\mathcal{L}(x, x+p) = \mathcal{L}(x+p) - \mathcal{L}(x) \\ \qquad \bar{\nabla}\mathcal{L}(x,x) = \nabla \mathcal{L}(x). \end{cases} \tag{12}$$

Discrete-Time Metric As in the analysis of continuous-time learning rules above, we define the negative *discrete* gradient as

$$\bar{y} \coloneqq -\bar{\nabla}\mathcal{L}(\theta_t, \theta_{t+1}). \tag{13}$$

Note that (10) and (12) together imply that updates of the parameter vector θ will correlate with $\bar{y}$

$$\eta\, g^\top \bar{y} \ = \ -\left[\mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t)\right] \ > \ 0.$$

Using this observation, we define the discrete analog of the metric (6) as

$$\bar{M} \coloneqq \frac{\bar{y}\bar{y}^\top}{\bar{y}^\top g} + \sum_{i=1}^{D-1} u_i u_i^\top \tag{14}$$

where as before, the vectors u_i are chosen to span the subspace orthogonal to g, denoted by $g^\perp \coloneqq \{v \in \mathbb{R}^n : v^\top g = 0\}$. We can see from this definition of $\bar{M}$ that

$$\bar{M} g = \bar{y} \qquad \text{and} \qquad \bar{M} = \bar{M}^\top \succ 0.$$

This implies, via (10), that the parameter updates can be written in the form of a positive definite matrix multiplied by the discrete gradient, as claimed

$$\theta_{t+1} = \theta_t - \eta \bar{M}^{-1}\, \bar{\nabla}\mathcal{L}\left(\theta_t,\, \theta_{t+1}\right). \tag{15}$$

Although (15) bears a resemblance to the natural gradient descent rule, they are not identical. This is because the discrete gradient does not always correspond to the gradient of a specific loss function. In the following section, we explore the conditions under which (15) can be considered a "true" natural gradient descent. In order to do this, we introduce a new discrete gradient, derived from the Hessian of the loss function.

Small Learning Rate Regime Motivated by Taylor's Theorem, we now introduce the following *particular* discrete gradient

$$\bar{\nabla}\mathcal{L}(x, x+p) \coloneqq \nabla \mathcal{L}(x) + \frac{1}{2} \nabla^2 \mathcal{L}(x + \lambda p)\, p \tag{16}$$

where $\lambda \in (0,1)$ is derived from (11). It can be easily verified that the discrete gradient conditions (12) hold. Taking $p = \eta\, g(\theta_t)$, we see that for $\bar{y}$ as in (13),

$$\bar{y} = -\nabla \mathcal{L}(\theta_t) - \frac{\eta}{2} \nabla^2 \mathcal{L}(\theta_t + \lambda \eta g)\, g = -\nabla \mathcal{L}(\theta_t) - \eta H g$$

where

$$H \coloneqq \frac{1}{2}\nabla^2\mathcal{L}(\theta_t + \lambda\eta g).$$

Note that for this particular choice of discrete gradient, we also have that

$$\bar{y} \to y \quad \text{as} \quad \eta \to 0.$$

Since (15) can be re-written as $\theta_{t+1} - \theta_t = \eta\overline{M}^{-1}\overline{y}$, from (16) we obtain

$$\bar{M}\left(\frac{\theta_{t+1} - \theta_t}{\eta}\right) = \bar{y} = -\nabla\mathcal{L}(\theta_t) - H\big(\theta_{t+1} - \theta_t\big).$$

Adding the Hessian term to both sides, we have

$$\big[\bar{M} + \eta H\big]\left(\frac{\theta_{t+1} - \theta_t}{\eta}\right) = -\nabla\mathcal{L}(\theta_t). \tag{17}$$

Equation (17) is almost in the desired natural gradient form. In order to put it in exactly natural gradient form, we would like the matrix $\bar{M} + H$ to be positive definite. We will now show that this can be done by choosing η sufficiently small. In the case where the loss function $\mathcal{L}$ is convex, $\bar{M} + H$ is always positive definite. We therefore only deal with the case when the loss $\mathcal{L}$ is non-convex, so that H has a negative minimum eigenvalue. That is, we assume

$$\exists\, h > 0 \quad \text{such that} \quad \lambda_{\min}(H) = -h.$$

Using the results of section 2.1 on picking a metric with an easily calculable minimum eigenvalue, and the fact [11] that

$$\lambda_{\min}(\bar{M} + \eta H) \geq \lambda_{\min}(\bar{M}) + \eta\lambda_{\min}(H)$$

we can ensure that $\lambda_{\min}(\bar{M} + \eta H) > 0$ by choosing η to be sufficiently small:

$$\eta < \frac{1}{h}\frac{\|\bar{y}\|}{\|g\|}\cos(\bar{\psi})$$

where $\bar{\psi}$ is the angle between the discrete gradient $\bar{y}$ and the update vector g, and is always between $-\pi/2$ and $\pi/2$. If η satisfies this inequality, then we can invert the $\bar{M} + \eta H$ term, yielding

$$\frac{\theta_{t+1} - \theta_t}{\eta} = -\big[\bar{M} + \eta H\big]^{-1}\nabla\mathcal{L}(\theta_t) \tag{18}$$

which is precisely a discrete-time natural gradient update rule.

Limit as Learning Rate Goes to Zero ($\eta \to 0$) Using the fact that

$$\bar{M} \to M \quad \text{and} \quad \eta H \to 0 \quad \text{as} \quad \eta \to 0,$$

we obtain that the limit of (18) as $\eta \to 0$ is

$$\dot{\theta} = -M^{-1}\ \nabla\mathcal{L}(\theta).$$

Stochastic Learning Rules When the discrete learning rule is stochastic, there is a probability distribution over θ_{t+1} given a known θ_t. In this case, the average update will be given as

$$\eta\langle g(\theta_t)\rangle = \langle\theta_{t+1} - \theta_t\rangle = \langle\theta_{t+1}\rangle - \theta_t.$$

Effective learning on average for a given loss $\mathcal{L}$, up to the generality of integrating this in time, can be defined as any learning rule that yields

$$\langle\mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t)\rangle < 0.$$

Similar to the deterministic case, we can define

$$\bar{M} = \frac{\langle\bar{y}\rangle\langle\bar{y}\rangle^\top}{\langle g\rangle^\top\langle\bar{y}\rangle} + \sum_{i=1}^{D-1} u_i u_i^\top$$

where $\bar{y}$ is the negative discrete gradient defined previously and vectors u_i span the subspace orthogonal to $\langle g \rangle$. This yields

$$\bar{M}\langle g \rangle = -\nabla L(\theta_t) - \eta \langle Hg \rangle.$$

We want the matrix

$$M \coloneqq \bar{M} + \eta \frac{\langle Hg \rangle \langle Hg \rangle^\top}{\langle Hg \rangle^\top \langle g \rangle} \tag{19}$$

to be positive definite to allow the average learning rule to be expressed as natural gradient descent,

$$\langle g \rangle = M^{-1} \nabla L(\theta_t). \tag{20}$$

Similar to the deterministic case, this will always hold for small enough η. This is because the second term in (19) can be made arbitrarily small as $\eta \to 0$.

3 Applications

3.1 Numerical Experiments

We provide two numerical experiments supporting the theory developed above. In the first, we show that a stable linear time-invariant (LTI) dynamical system, which in general cannot be written as the gradient of a scalar function, can be written in the natural gradient form. In the second, we show that a popular biologically-plausible alternative to propagation, Feedback Alignment, can also be written as a natural gradient descent.

Linear Time-Invariant Dynamics We consider the stable LTI system

$$\dot{\theta} = g(\theta, t) = A\theta \tag{21}$$

where A is an asymmetric matrix with eigenvalues strictly in the left-hand side of the complex plane. Because A is asymmetric, the dynamics (21) cannot be written as the gradient of a scalar function (because this would imply the Hessian, A, is symmetric). Of course, it is well-know that the trajectories of (21) *do* decrease the Lyapunov function

$$\mathcal{L}(\theta) = \theta^\top P \theta \qquad \text{where} \qquad PA + A^\top P = -Q$$

with $Q = Q^\top$, $P = P^\top \succ 0$ [27]. In simulations, we set $Q = I$ and solved for P by using the SciPy [28] function `scipy.linalg.solve_continuous_lyapunov`. In this case, we have that

$$y = -\nabla_\theta \mathcal{L} = -2P\theta$$

and the metric M can be calculated according to our results, putting the dynamics (21) in the natural gradient form

$$\dot{\theta} = -M^{-1}(\theta)\, \nabla_\theta \mathcal{L}$$

Figure 3 shows the results.

Biologically Plausible Learning (Feedback Alignment) Feedback alignment (FA) is a biologically plausible alternative to backpropagation (BP) [6] with strong performance on benchmarks and favorable scaling for large networks [29, 30]. FA uses a random, fixed backward connectivity structure instead of BP's symmetric weights. We train a simple linear network on MNIST, and FA, as expected, improves performance. We also derive the metric M that relates the updates of BP to FA. The results can be found in Figure 3.

4 Discussion

Contributions and Related Work It is known that if a continuous-time dynamical system has a strict Lyapunov function, the system can be described by a symmetric positive definite matrix multiplied by the negative gradient of the Lyapunov function [26, 31]. In our case, the loss function acts as the Lyapunov function for learning dynamics.

Our work extends the results of McLachlan et al. [26] to both continuous-time and discrete-time (both deterministic and stochastic) systems, even when the loss does not decrease monotonically and is potentially time-varying. We prove that the class of metrics considered in McLachlan et al. [26] and Bárta et al. [31] is canonical–it is the *only* class of valid metrics. We derive a one-parameter family of metrics for which the spectrum can be calculated exactly. We show that the "optimal" metric (in terms of having the smallest condition number) over all possible metrics exists within this one-parameter family. This proves to be especially useful in the analysis of discrete-time learning rules.

Conceptually, our findings help clarify discussions about learning rules by showing that effective learning rules, including biological ones, belong to the class of natural gradient algorithms. This applies in both continuous and discrete time, supporting the idea that the gradient is fundamental to all learning processes.

Limitations & Future Work We conjecture that any sequence of parameter updates leading to overall improvement in a loss function (even if not monotonically) can be reformulated as natural gradient descent for some appropriately chosen loss function and metric. Future work will focus on broadening the scope of the results presented here, aiming to prove this conjecture in full generality.

Acknowledgements

LK acknowledges Professor Jean-Jacques Slotine (MIT) for helpful conversations.

A Optimal Metric

Eigenvalues of One-Parameter Family For convenience, we begin by setting $\alpha = \gamma \frac{y^\top y}{y^\top g}$, for some $\gamma > 0$.

Lemma 1. *The eigenvalues of*

$$M = \frac{yy^\top}{y^\top g} + \gamma \frac{y^\top y}{y^\top g}\left(I - \frac{gg^\top}{g^\top g}\right) \tag{22}$$

are

$$\lambda_{\max\,/\,\min}(M) = \frac{\|y\|}{2\|g\|\cos(\psi)}\left((1+\gamma) \pm \sqrt{(1+\gamma)^2 - 4\gamma\cos^2(\psi)}\right),$$

with multiplicity one each, and

$$\frac{\|y\|}{\|g\|\cos(\psi)}\gamma$$

with multiplicity $D-2$.

Proof. Note that

$$M = \frac{\|y\|}{\|g\|\cos(\psi)}\left(\hat{y}\hat{y}^\top + \gamma(I - \hat{g}\hat{g}^\top)\right).$$

Thus it suffices to compute the eigenvalues of

$$A_0 := \hat{y}\hat{y}^\top - \gamma\hat{g}\hat{g}^\top.$$

Then for $v = \hat{y} + \zeta\hat{g}$, we have

$$A_0 v = \big(\hat{y} - \cos(\psi)\gamma\hat{g}\big) + \zeta\big(\cos(\psi)\hat{y} - \gamma\hat{g}\big) = \big(1 + \zeta\cos(\psi)\big)\hat{y} - \big(\cos(\psi)\gamma + \zeta\gamma\big)\hat{g}.$$

This is a multiple of v exactly when

$$\zeta\big(1 + \zeta\cos(\psi)\big) = -\big(\cos(\psi)\gamma + \zeta\gamma\big),$$

which gives $\cos(\psi)\zeta^2 + (1+\gamma)\zeta + \gamma\cos(\psi) = 0$, i.e.,

$$\zeta = \frac{1}{2\cos(\psi)}\left(-(\gamma+1) \pm \sqrt{(\gamma+1)^2 - 4\gamma\cos^2(\psi)}\right).$$

Thus the corresponding eigenvalue of A_0 is

$$\lambda_{\max / \min}(A_0) = 1 + \zeta \cos(\psi) = \frac{1}{2}\left(1 - \gamma \pm \sqrt{(\gamma+1)^2 - 4\gamma\cos^2(\psi)}\right).$$

They correspond to the eigenvalues

$$\lambda_{\max / \min}(M) = \frac{\|y\|}{2\|g\|\cos(\psi)}\left(1 + \gamma \pm \sqrt{(\gamma+1)^2 - 4\gamma\cos^2(\psi)}\right). \qquad \square$$

Optimal Condition Number for One-Parameter Family The condition number for M as in (22) is

$$\kappa(M) = \frac{1+\gamma+\sqrt{(\gamma+1)^2 - 4\gamma\cos^2(\psi)}}{1+\gamma-\sqrt{(\gamma+1)^2 - 4\gamma\cos^2(\psi)}} = \frac{\left(1+\gamma+\sqrt{(\gamma+1)^2 - 4\gamma\cos^2(\psi)}\right)^2}{4\gamma\cos^2(\psi)}.$$

so

$$\sqrt{\kappa(M)} = \frac{1+\gamma+\sqrt{(\gamma+1)^2 - 4\gamma\cos^2(\psi)}}{2\sqrt{\gamma}\cos(\psi)} = \frac{\gamma^{1/2} + \gamma^{-1/2} + \sqrt{(\gamma^{1/2}+\gamma^{-1/2})^2 - 4\cos^2(\psi)}}{2\cos(\psi)}.$$

Now let $\cos(\psi)z = \gamma^{1/2} + \gamma^{-1/2}$, which is some variable $z \geq 2/\cos(\psi)$. We are trying to minimize

$$\sqrt{\kappa(M)} = \frac{z + \sqrt{z^2-4}}{2}.$$

This is a monotonically increasing function of z so is minimized at $z = 2/\cos(\psi)$. This corresponds to $\gamma = 1$.

Optimal Condition Number for all Metrics Let us prove a lemma:

Lemma 2. *Suppose v_0, v_1 are vectors. The following are equivalent:*

1. *for any B a symmetric matrix, $v_0^\top B v_0 = v_1^\top B v_1$; and*
2. *$v_0 = \pm v_1$.*

Proof. One direction is obvious. For the non-obvious direction, let $B = (b_{ij})_{i,j=1}^D$ where $b_{ij} = b_{ji}$. Let $v_0 = (x_1, \ldots, x_D)^\top$ and $v_1 = (y_1, \ldots, y_D)^\top$. Then we have the equation

$$\sum_{i,j=1}^D b_{ij} x_i x_j = \sum_{i,j=1}^D b_{ij} y_i y_j.$$

Thus we conclude that $x_i x_j = y_i y_j$ for any two indices i and j. When $i = j$ this implies $x_i = \pm y_i$, but these signs must all be the same using all the other equations. $\square$

As a consequence, we can prove the following variant:

Lemma 3. *Suppose v_0 and v_1 are vectors, and g is a non-zero vector. The following are equivalent:*

1. *for any symmetric matrix B such that $Bg = 0$, the equality $v_0^\top B v_0 = v_1^\top B v_1$ holds; and*
2. *there exists an $\alpha \in \mathbb{R}$ such that $v_0 = \pm v_1 + \alpha g$.*

Proof. Again, one direction is obvious. For the non-obvious direction, we see consider the projection of v_0 and v_1 to $g^\perp$ along g:

$$v_0' = v_0 - \frac{g^\top v_0}{g^\top g} g, \qquad v_1' = v_1 - \frac{g^\top v_1}{g^\top g} g.$$

Then we see that $(v_0')^\top B v_0' = (v_1')^\top B v_1'$ for all symmetric matrices B on $g^\perp$. Now using Lemma 2 we see that $v_0' = \pm v_1'$. $\square$

Proposition 1. *Let M be a positive definite matrix such that $Mg = y$ where $y^\top g > 0$. Let ψ be the angle between g and y. Then the minimum value for $\kappa(M)$, achieved by (22) when $\gamma = 1$, is*

$$\frac{1 + |\sin(\psi)|}{1 - |\sin(\psi)|}.$$

Proof. We know that $M = \frac{1}{y^\top g} yy^\top + M'$ for some symmetric positive definite matrix M' on $g^\perp$. Let M' be a matrix attaining the minimum $\kappa(M)$. Consider the perturbation of M' by some symmetric matrix B such that $Bg = 0$. Then perturbation theory tells us

$$\lambda_{\max / \min}(M + \epsilon B) = \lambda_{\max / \min}(M) + v_{\max / \min}^\top B v_{\max / \min} \epsilon + O(\epsilon^2)$$

where $v_{\max / \min}$ are eigenvectors of M with eigenvalue $\lambda_{\max / \min}$ normalized to have norm 1. Thus

$$\frac{\lambda_{\max}(M + \epsilon B)}{\lambda_{\min}(M + \epsilon B)} = \frac{\lambda_{\max}(M) + v_{\max}^\top B v_{\max} \epsilon}{\lambda_{\min}(M) + v_{\min}^\top B v_{\min} \epsilon} + O(\epsilon^2).$$

Since M' is in particular a local minimum,

$$\lambda_{\max}(M) \cdot v_{\min}^\top B v_{\min} = \lambda_{\min}(M) \cdot v_{\max}^\top B v_{\max}.$$

By Lemma 3 this implies that $v_{\max}$, $v_{\min}$, and g are linearly dependent. Hence by applying M, we see that so is $v_{\max}$, $v_{\min}$, and y. So we can restrict ourself to this two-dimensional subspace spanned by $v_{\max}$ and $v_{\min}$. But then the same argument as before shows optimality. □

Optimal Maximum and Minimum Metric Eigenvalues We show that a lower bound (resp., upper bound) for $\lambda_{\min}(M)$ (resp., $\lambda_{\max}(M)$) for matrices M satisfying the conditions of Proposition 1 and show that they are never achieved but are asymptotically achieved.

Proposition 2. *Let M be a symmetric positive definite matrix such that $Mg = y$ where $y^\top g > 0$. Let ψ be the angle between g and y. Then $\lambda_{\min}(M) < \frac{\|y\|}{\|g\|} \cos(\psi)$. Moreover, the supremum is asymptotically approached by (22) as $\gamma \to \infty$.*

Proof. Recall that

$$\lambda_{\min}(M) = \min_{v \neq 0} \frac{v^\top M v}{v^\top v}, \tag{23}$$

and moreover the minimum is reached by eigenvectors with eigenvalue $\lambda_{\min}$. Thus

$$\lambda_{\min}(M) \leq \frac{g^\top M g}{g^\top g} = \frac{g^\top y}{g^\top g} = \frac{\|y\|}{\|g\|} \cos(\psi).$$

Moreover, equality is not reached since g is not an eigenvector of M. Finally, the minimum eigenvalue of (22) as $\gamma \to \infty$ is, by Lemma 1,

$$\lim_{\gamma \to \infty} \frac{\|y\|}{2\|g\| \cos(\psi)} \left((1 + \gamma) - \sqrt{(1 + \gamma)^2 - 4\gamma \cos^2(\psi)} \right) = \frac{\|y\|}{\|g\|} \cos(\psi).$$ □

Proposition 3. *Let M be a symmetric positive definite matrix such that $Mg = y$ where $y^\top g > 0$. Let ψ be the angle between g and y. Then $\lambda_{\max}(M) > \frac{\|y\|}{\|g\| \cos(\psi)}$. Moreover, the infimum is asymptotically approached by (22) as $\gamma \to 0$.*

Proof. Recall that (e.g., by using (23) and observing that $\lambda_{\max}(M) = \lambda_{\min}(M^{-1})^{-1}$)

$$\lambda_{\max}(M) = \max_{v \neq 0} \frac{v^\top v}{v^\top M^{-1} v},$$

and moreover the minimum is reached by the eigenvectors with eigenvalue $\lambda_{\max}$. Thus

$$\lambda_{\max}(M) \leq \frac{y^\top y}{y^\top M^{-1} y} = \frac{y^\top y}{y^\top g} = \frac{\|y\|}{\|g\| \cos(\psi)}.$$

Moreover, equality is not reached since y is not an eigenvector of M. Finally, the maximum eigenvalue of (22) as $\gamma \to 0$ is, by Lemma 1,

$$\lim_{\gamma \to 0} \frac{\|y\|}{2\|g\| \cos(\psi)} \left((1 + \gamma) + \sqrt{(1 + \gamma)^2 - 4\gamma \cos^2(\psi)} \right) = \frac{\|y\|}{\|g\| \cos(\psi)}.$$ □

References

[1] Colin Bredenberg and Cristina Savin. Desiderata for normative models of synaptic plasticity. *Neural Computation*, 36(7):1245–1285, 2024.

[2] Blake Aaron Richards and Konrad Paul Kording. The study of plasticity has always been about gradients. *The Journal of Physiology*, 601(15):3141–3149, 2023.

[3] Augustin Cauchy et al. Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538, 1847.

[4] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 1999.

[5] Simone Carlo Surace, Jean-Pascal Pfister, Wulfram Gerstner, and Johanni Brea. On the choice of metric in gradient-based theories of brain function, December 2018.

[6] Timothy P Lillicrap, Daniel Cownden, Douglas B Tweed, and Colin J Akerman. Random synaptic feedback weights support error backpropagation for deep learning. *Nature communications*, 7(1):13276, 2016.

[7] Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10 (2):251–276, 1998.

[8] Michael Muehlebach and Michael Jordan. A dynamical systems perspective on nesterov acceleration. In *International Conference on Machine Learning*, pages 4656–4662. PMLR, 2019.

[9] Amir Ali Ahmadi and Pablo A Parrilo. Non-monotonic lyapunov functions for stability of discrete time nonlinear and switched systems. In *2008 47th IEEE conference on decision and control*, pages 614–621. IEEE, 2008.

[10] Frank H Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205:247–262, 1975.

[11] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[12] Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

[13] R Pascanu. Revisiting natural gradient for deep networks. *arXiv preprint arXiv:1301.3584*, 2013.

[14] James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International conference on machine learning*, pages 2408–2417. PMLR, 2015.

[15] James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

[16] Felix Dangel, Johannes Müller, and Marius Zeinhofer. Kronecker-factored approximate curvature for physics-informed neural networks. *arXiv preprint arXiv:2405.15603*, 2024.

[17] Patrick M. Wensing and Jean-Jacques E. Slotine. Beyond Convexity – Contraction and Global Convergence of Gradient Descent. *arXiv:1806.06655 [math]*, August 2020.

[18] Yann Ollivier, Ludovic Arnold, Anne Auger, and Nikolaus Hansen. Information-geometric optimization algorithms: A unifying picture via invariance principles. *Journal of Machine Learning Research*, 18(18):1–65, 2017.

[19] Taeyoon Lee, Jaewoon Kwon, and Frank C Park. A natural adaptive control law for robot manipulators. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9. IEEE, 2018.

[20] Nicholas M Boffi and Jean-Jacques E Slotine. Implicit regularization and momentum algorithms in nonlinearly parameterized adaptive control and prediction. *Neural Computation*, 33(3):590–673, 2021.

[21] Belinda Tzen, Anant Raj, Maxim Raginsky, and Francis Bach. Variational principles for mirror descent and mirror langevin dynamics. *IEEE Control Systems Letters*, 2023.

[22] Roman Pogodin, Jonathan Cornford, Arna Ghosh, Gauthier Gidel, Guillaume Lajoie, and Blake Richards. Synaptic Weight Distributions Depend on the Geometry of Plasticity, May 2023.

[23] Elad Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

[24] Walter Rudin et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1964.

[25] Oscar Gonzalez. Time integration and discrete hamiltonian systems. *Journal of Nonlinear Science*, 6:449–467, 1996.

[26] Robert I McLachlan, G Reinout W Quispel, and Nicolas Robidoux. Geometric integration using discrete gradients. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 357(1754):1021–1045, 1999.

[27] Roger W Brockett. *Finite dimensional linear systems*. SIAM, 2015.

[28] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[29] Arild Nøkland. Direct feedback alignment provides learning in deep neural networks. *Advances in neural information processing systems*, 29, 2016.

[30] Julien Launay, Iacopo Poli, François Boniface, and Florent Krzakala. Direct feedback alignment scales to modern deep learning tasks and architectures. *Advances in neural information processing systems*, 33:9346–9360, 2020.

[31] Tomáš Bárta, Ralph Chill, and Eva Fašangová. Every ordinary differential equation with a strict lyapunov function is a gradient system. *Monatshefte für Mathematik*, 166(1):57–72, 2012.