Nature-Inspired Local Propagation

Alessandro Betti
IMT School for Advanced Studies
Lucca, Italy
alessandro.betti@imtlucca.it

Marco Gori
DIISM
University of Siena
Siena, Italy
marco.gori@unisi.it

Abstract

The spectacular results achieved in machine learning, including the recent advances in generative AI, rely on large data collections. On the opposite, intelligent processes in nature arises without the need for such collections, but simply by on-line processing of the environmental information. In particular, natural learning processes rely on mechanisms where data representation and learning are intertwined in such a way to respect spatiotemporal locality. This paper shows that such a feature arises from a pre-algorithmic view of learning that is inspired by related studies in Theoretical Physics. We show that the algorithmic interpretation of the derived "laws of learning", which takes the structure of Hamiltonian equations, reduces to Backpropagation when the speed of propagation goes to infinity. This opens the doors to machine learning studies based on full on-line information processing that are based on the replacement of Backpropagation with the proposed spatiotemporal local algorithm.

1 Introduction

By and large, the spectacular results of Machine Learning in nearly any application domain strongly rely on large data collections along with associated professional skills. Interestingly, the successful artificial schemes that we have been experimenting under this framework are far away from the solutions that Biology seems to have discovered. We have recently seen a remarkable effort in the scientific community to explore biologically inspired models (e.g. see [31, 16, 30, 18]) where the crucial role of temporal information processing it is clearly identified.

While this paper is related to those investigations, it is based on more strict assumptions on environmental interactions that might stimulate efforts towards a more radical transformation of machine learning with emphasis on the temporal domain. In particular, we assume that learning and inference develop jointly under a nature based protocol of environmental interactions and then we suggest developing computational learning schemes regardless of biological solutions. Basically, the agent is not given the privilege of recording the temporal stream, but only to represent it properly by appropriate abstraction mechanisms. While the agent can obviously use its internal memory for storing those representations, we assume that it cannot access data collection. Instead, the agent can only rely on buffers of limited size to retain the information it acquires. From a cognitive perspective, these small buffers allow the agent to review recent inputs backward in time, implementing a form of selective attention.

We propose a pre-algorithmic framework which derives from the formulation of learning as an Optimal Control problem [19] and propose an approach to its solution that is also inspired by principles of Theoretical Physics. We formulate the continuous learning problem to emphasize how optimization theory brings out solutions based on differential equations that recall similar laws in nature. The discrete counterpart [2, p. 2], which is more similar to recurrent neural network algorithms that are found in the literature, can be derived as a numerical method and applied in

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-1989

practical scenarios like lifelong learning with long video streams [4], where an Euler method for the differential equations can serve as an "optimizer" for RNN weights. Interestingly, we demonstrate that the online computation described in this paper achieves spatiotemporal locality, thereby contributing to the longstanding debate on the biological plausibility of Backpropagation [8, 33, 21]. Specifically, we address the *update locking problem* and the issue of infinitely fast signal propagation in neural networks. Finally, the paper shows that the conquest of locality opens up a fundamental problem, namely that of approximating the solution of Hamilton's equations with boundary conditions using only initial conditions. A few insights on the solution of this problem are given for the task of tracking in optimal control, which opens the doors of a massive investigation of the proposed approach.

2 Recurrent Neural Networks and spatiotemporal locality

We put ourselves in the general case where the computational model that we are considering is based on a digraph $D = (V, A)$ where $V = \{1, 2, \dots, n\}$ is the set of vertices and and A is the set of directed arches that defines the structure of the graph. Let $\mathrm{ch}(i)$ denote the set of vertices that are children of vertex i and with $\mathrm{pa}(i)$ the set of vertices that are parents of vertex i for any given $i \in V$. More precisely we are interested in the computation of neuron outputs over a temporal horizon $[0, T]$. Formally, this involves assigning each vertex $i \in V$ a trajectory $t \mapsto x_i(t)$ of outputs that is computed based on the outputs of other neurons and environmental information. The environmental information is mathematically represented by a trajectory[1] $u\colon [0, +\infty) \to \mathbb{R}^d$. We will assume that the output of the first d neurons (i.e the value of x_i for $i = 1, \dots, d$) matches the value of the components of the input: $x_i(t) = u_i(t)$ for $i = 1, \dots d$ and $\forall t \in [0, T]$. In order to consistently interpret the first d neurons as input we require two additional property of the graph structure:

$$\mathrm{pa}(i) = \emptyset \quad \forall i = 1, \dots, d; \tag{1}$$

$$\mathrm{pa}(\{d+1, \dots, n\}) \supset \{1, \dots, d\}. \tag{2}$$

Here (1) says that an input neuron do not have parents, and it also implies that no self loops are allowed for the input neurons. On the other hand (2) means that all input neurons are connected to at least one other neuron amongst $\{d+1, \dots, n\}$.

We will denote with $x(t)$ (without a subscript) the ordered list of all the output of the neurons at time t except for the input neurons, $x(t) := (x_{d+1}(t), \dots, x_n(t))$, and with this definition we can represent $x(t)$ for any $t \in [0, T]$ as a vector in the euclidean space $\mathbb{R}^{n-d}$. This vector is usually called the *state* of the network since its knowledge gives you the precise value of each neuron in the net. The parameters of the model are instead associated to the arcs of the graph via the map $(j, i) \in A \mapsto w_{ij}$ where w_{ij} assumes values on $\mathbb{R}$. We will denote with $w_{i*}(t) \in \mathbb{R}^{|\mathrm{pa}(i)|}$ the vector composed of all the weights corresponding to arches of the form (j, i). If we let $N := \sum_{i=1}^{n} |\mathrm{pa}(i)|$ the total number of weights of the model we also define $\mathbb{R}^N \ni \boldsymbol{w}(t) := (w_{1*}(t), \dots, w_{n*}(t))$ the concatenation of all the weights of the network. Finally we will assume that the output of the model is computed in terms of a subset of the neurons. More precisely we will assume that, given a vector of m indices $(i_1, \dots, i_m)$ with $i_k \in \{d+1, \dots, n\}$, at each temporal instant the output of the net is a function $\pi\colon \mathbb{R}^m \to \mathbb{R}^h$ of $(x_{i_1}, \dots, x_{i_m})$. For future convenience we will denote $O = \{i_1, \dots, i_m\}$.

Temporal locality and causality In general we are interested in computational schemes which are both local in time and causal. Let us assume that we are working at some fixed temporal resolution τ, meaning that we can define a partition of the half line $(0, +\infty)$, $\mathcal{P} := \{0 = t_\tau^0 < t_\tau^1 < \cdots < t_\tau^n < \dots\}$ with $t_\tau^n = t_\tau^{n-1} + \tau$, then the input signal becomes a sequence of vectors $(U_\tau^n)_{n=0}^{+\infty}$ with $U_\tau^n := u(t_\tau^n)$ and the neural outputs and parameters can be regarded as an approximation of the trajectories x and $\boldsymbol{w}$: $X_\tau^n \approx x(t_\tau^n)$ and $W_\tau^n \approx \boldsymbol{w}(t_\tau^n)$, $n = 1, \dots \lfloor T/\tau \rfloor$. A local computational rule for the neural outputs means that X_τ^n is a function of $X_\tau^{n-l}, \dots, X_\tau^n, \dots, X_\tau^{n+l}, W_\tau^{n-l}, \dots, W_\tau^n, \dots, W_\tau^{n+l}$ and

[1] In the reminder of the paper we will try whenever possible to formally introduce functions by clearly stating domain and co-domain. In particular whenever the function acts on a product space we will try to use a consistent notation for the elements in the various sets that define the input so that we can re-use such notation to denote the partial derivative of such function. For instance let us suppose that $f\colon A \times B \to \mathbb{R}$ is a function that maps $(a, b) \mapsto f(a, b)$ for all $a \in A$ and $b \in B$. Then we will denote with f_a *the function* that represents the partial derivative of f with respect to its first argument, with f_b the partial derivative of f with respect to its second argument *as a function* and so on. We will instead denote, for instance, with $f_a(x, y)$ the element of $\mathbb{R}$ that represent the value of f_a on the point $(x, y) \in A \times B$.

$t_\tau^{n-l}, \dots, t_\tau^n, \dots, t_\tau^{n+l}$, where $l \ll T/\tau$ can be thought as the order of locality. If we assume that $l \equiv 1$ (first order method) then

$$X_\tau^n = F(X_\tau^{n-1}, X_\tau^n, X_\tau^{n+1}, W_\tau^{n-1}, W_\tau^n, W_\tau^{n+1}, t_\tau^{n-1}, t_\tau^n, t_\tau^{n+1}). \tag{3}$$

Causality instead expresses the fact that only past information can influence the current state of the variables meaning that actually (3) should be replaced by $X_\tau^n = F(X_\tau^{n-1}, W_\tau^{n-1}, t_\tau^{n-1})$. Returning to the continuous description, this equation can be interpreted as a discretization of a Cauchy problem for

$$\dot{x} = f(x, \boldsymbol{w}, t), \tag{4}$$

with assigned initial conditions on $x(0)$. Note that the ability to determine the solution by evolving the state from a specified initial value is fundamentally due to our causality requirement.

Spatial locality Furthermore we assume that such computational scheme is local in time and make use only on spatially local (with respect to the structure of the graph) quantities as follows:

$$\begin{cases} x_i(t) = u_i(t) & \text{for } i = 1, \dots d \quad \text{and} \quad \forall t \in [0, T]; \\ c_i^{-1}\dot{x}_i(t) = \Psi^i(x_i(t), \textsc{pa}^i(x(t)), \textsc{in}^i(\boldsymbol{w}(t))) & \text{for } i = d+1, \dots, n \quad \text{and} \quad \forall t \in [0, T]. \end{cases} \tag{5}$$

Here $c_i > 0$ for all $i = d+1, \dots, n$ sets the velocity constant that controls the updates of the i-th neuron, $\Psi^i \colon \mathbb{R} \times \mathbb{R}^{|\operatorname{pa}(i)|} \times \mathbb{R}^{|\operatorname{pa}(i)|} \to \mathbb{R}$ for all $i = d+1, \dots, n$ performs the mapping $(r, \alpha, \beta) \mapsto \Psi^i(r, \alpha, \beta)$ for all $r \in \mathbb{R}$, $\alpha, \beta \in \mathbb{R}^{|\operatorname{pa}(i)|}$, $\textsc{pa}^i \colon \mathbb{R}^{n-d} \to \mathbb{R}^{|\operatorname{pa}(i)|}$ project the vector $\xi \in \mathbb{R}^{n-d} \mapsto \textsc{pa}^i(\xi)$ on the subspace generated by neurons which are in $\operatorname{pa}(i)$ and $\textsc{in}^i \colon \mathbb{R}^N \to \mathbb{R}^{|\operatorname{pa}(i)|}$ maps the any vector $\boldsymbol{\omega} \in \mathbb{R}^N \mapsto \textsc{in}^i(\boldsymbol{\omega})$ onto the space spanned by only the weights associated to arcs that points to neuron i. The assumptions summarized above describe the basic properties of a RNN or, as sometimes is referred to when dealing with a continuous time computation, a Continuous Time RNN [32]. The typical form of function Ψ_i, is the following

$$\Psi^i(r, \alpha, \beta) = -r + \sigma(\beta \cdot \alpha), \quad \forall r \in \mathbb{R} \quad \text{and} \quad \forall \alpha, \beta \in \mathbb{R}^{|\operatorname{pa}(i)|}. \tag{6}$$

where in this case $\cdot$ is the standard scalar product on $\mathbb{R}^{|\operatorname{pa}(i)|}$ and $\sigma \colon \mathbb{R} \to \mathbb{R}$ is a nonlinear bounded smooth function (usually a sigmoid-like activation function). Under this assumption the state equation in (5) becomes

$$c_i^{-1}\dot{x}_i(t) = -x_i(t) + \sigma(\textsc{in}^i(\boldsymbol{w}(t)) \cdot \textsc{pa}^i(x(t))) \equiv -x_i(t) + \sigma\Big(\sum_{j \in \operatorname{pa}(i)} w_{ij} x_j(t)\Big), \tag{7}$$

which is indeed the classical neural computation. Here we sketch a result on the Bounded Input Bounded Output (BIBO) stability of this class of recurrent neural network which is also important for the learning process that will be described later.

Propostion 1. *The recurrent neural network defined by ODE (7) is (BIBO) stable.*

Proof. See Appendix D □

3 Learning as a Variational Problem

In the computational model described in Section 2, once the graph D and an input u are assigned, the dynamics of the model is determined solely by the functions that describes the changes of the weights over time. Inspired by the Cognitive Action Principle [3] that formulate learning for FNN in terms of a variational problem, we claim that in an online setting the laws of learning for recurrent architectures can also be characterized by minimality of a class of functional. In what follows we will then consider variational problems for a functional of the form

$$F(\boldsymbol{w}) = \int_0^T \left[\frac{mc}{2}|\dot{\boldsymbol{w}}|^2 + c\ell(\boldsymbol{w}(t), x(t; \boldsymbol{w}), t)\right]\phi(t)\, dt, \tag{8}$$

where $x(\cdot, \boldsymbol{w})$ is the solution of (4) with fixed initial conditions[2], $\phi \colon [0, T] \to \mathbb{R}$ is a strictly positive smooth function that weights the integrand, $m > 0$, $\ell \colon \mathbb{R}^n \times \mathbb{R}^N \times [0, T] \to \overline{\mathbb{R}}_+$ is a positive function and finally $c := \sum_{i=d+1}^n c_i/(n-d)$. We discuss the requirements for making the stationarity conditions of this class of functional both temporally and spatially local and how they can be interpreted as learning rules.

[2]We do not explicitly indicate the dependence on the initial condition to avoid cumbersome notation.

https://doi.org/10.52202/079017-1989

3.1 Optimal Control Approach

The problem of minimizing the functional in (8) can be solved by making use of the formalism of Optimal Control. The first step is to put this problem in the canonical form by introducing an additional control variable as follow

$$G(\boldsymbol{v}) = \int_0^T \left[\frac{mc}{2}|\boldsymbol{v}|^2 + c\ell(\boldsymbol{w}(t;\boldsymbol{v}), x(t;\boldsymbol{v}), t) \right] \phi(t)\, dt, \tag{9}$$

where $\boldsymbol{w}(t;\boldsymbol{v})$ and $x(t;\boldsymbol{v})$ solve

$$\dot{x}(t) = f(x(t), \boldsymbol{w}(t), t), \quad \text{and} \quad \dot{\boldsymbol{w}}(t) = \boldsymbol{v}(t). \tag{10}$$

Then, the minimality conditions can be expressed in terms of the Hamiltonian function (see Appendix A), that is defined for every $\xi \in \mathbb{R}^N$, $\boldsymbol{\omega} \in \mathbb{R}^n$, $p \in \mathbb{R}^N$, $q \in \mathbb{R}^n$ and $t \in [0,T]$ as:

$$H(\xi, \boldsymbol{\omega}, p, q, t) = -\frac{1}{\phi(t)} \frac{q^2}{2mc} + c\ell(\boldsymbol{\omega}, \xi, t)\phi(t) + p \cdot f(\xi, \boldsymbol{\omega}, t), \tag{11}$$

via the following general result.

Theorem 1 (Hamilton equations)**.** *Let H be as in* (11) *and assume that $x(0) = x^0$ and $\boldsymbol{w}(0) = \boldsymbol{w}^0$ are given. Then a minimum of the functional in* (9) *satisfies the Hamilton equations:*

$$\begin{cases} \dot{x}(t) = f(x(t), \boldsymbol{w}(t), t) \\ \dot{\boldsymbol{w}}(t) = -p_{\boldsymbol{w}}(t)/(mc\phi(t)) \\ \dot{p}_x(t) = -p_x(t) \cdot f_\xi(x(t), \boldsymbol{w}(t), t) - c\ell_\xi(\boldsymbol{w}(t), x(t), t)\phi(t) \\ \dot{p}_{\boldsymbol{w}}(t) = -p_x(t) \cdot f_{\boldsymbol{\omega}}(x(t), \boldsymbol{w}(t), t) - c\ell_{\boldsymbol{\omega}}(\boldsymbol{w}(t), x(t), t)\phi(t) \end{cases} \tag{12}$$

together with the boundary conditions

$$p_x(T) = p_{\boldsymbol{w}}(T) = 0. \tag{13}$$

Proof. See Appendix A. □

3.2 Recovering spatio-temporal locality

Starting from the general expressions for the stationarity conditions expressed by (12) and (13), we will now discuss how the temporal and spatial locality assumptions that we made on our computational model in Section 2 leads to spatial and temporal locality of the update rules of the parameters $\boldsymbol{w}$.

Temporal Locality The local structure of (10), that comes from the locality of the computational model that we discussed in Section 2 guarantees the locality of Hamilton's equations 12. However the functional in (9) has a global nature (it is an integral over the whole temporal interval) and the differential term $m|\boldsymbol{v}|^2/2$ links the value of the parameters across near temporal instant giving rise to boundary conditions in (13). This also means that, strictly speaking (12) and (13) overall define a problem that is non-local in time. We will devote the entire Section 4 to discuss this central issue.

Spatial Locality The spatial locality of (12) directly comes from the specific form of the dynamical system in (5) and from a set of assumptions on the form of the term ℓ. In particular we have the following result:

Theorem 2. *Let $\ell(\boldsymbol{\omega}, \xi, s) = kV(\boldsymbol{\omega}, s) + L(\xi, s)$ for every $(\boldsymbol{\omega}, \xi, s) \in \mathbb{R}^N \times \mathbb{R}^{n-d} \times [0,T]$, where $V\colon \mathbb{R}^N \times [0,T] \to \overline{\mathbb{R}}_+$ is a regularization term on the weights*[3] *and $L\colon \mathbb{R}^{n-d} \times [0,T] \to \overline{\mathbb{R}}_+$ depends only on the subset of neurons from which we assume the output of the model is computed, that is $L_{\xi_i}(\xi, s) = L_{\xi_i}(\xi, s)1_O(i)$, where 1_O is the indicator function of the set of the output neurons. Let Ψ^i be as in* (6) *for all $i = d+1, \dots, n$, then the generic Hamilton's equations described in* (12) *become*

$$\begin{cases} c_i^{-1}\dot{x}_i = -x_i + \sigma\Big(\sum_{j \in \mathrm{pa}(i)} w_{ij} x_j\Big) \\ \dot{w}_{ij} = -p^{ij}_{\boldsymbol{w}}/(mc\phi) \\ \dot{p}^i_x = c_i p^i_x - \sum_{k \in \mathrm{ch}(i)} c_k \sigma'\Big(\sum_{j \in \mathrm{pa}(k)} w_{kj} x_j\Big) p^k_x w_{ki} - cL_{\xi_i}(x,t)\phi \\ \dot{p}^{ij}_{\boldsymbol{w}}(t) = -c_i p^i_x \sigma'\Big(\sum_{m \in \mathrm{pa}(i)} w_{im} x_m\Big) x_j - ckV_{\boldsymbol{\omega}_{ij}}(\boldsymbol{w}, t)\phi \end{cases} \tag{14}$$

[3] a typical choice for this function could be $V(\boldsymbol{\omega}, s) = |\boldsymbol{\omega}|^2/2$ with $k > 0$

Proof. See Appendix B. □

Remark 1. Notice (14) directly inherit the spatially local structure from the assumption in (5).

Theorem 2 other than giving us spatio-temporal rules show that the computation of the p_x has a very distinctive and familiar property: for each neuron the values of p_x^i are computed using quantities defined on children's nodes as it happens for the computations of the gradients in the Backpropagation algorithm for a FNN. In order to better understand Eq. (14) let us define an appropriately normalized costate

$$\lambda_x^i(t) := \frac{\sigma'(a_i(t))}{\phi(t)} p_x^i(t), \quad \text{with} \quad a_i(t) = \sum_{m \in \mathrm{pa}(i)} w_{im} x_m \quad \forall i = d+1, \dots, n, \tag{15}$$

where we have introduced the notation $a_i(t)$ to stand for the activation of neuron i.[4] With these definitions we are ready to state the following result

Propostion 2. *The differential system in* (14) *is equivalent to the following system of ODE of mixed orders:*

$$\begin{cases} c_i^{-1} \dot{x}_i = -x_i + \sigma(a_i); \\ \ddot{w}_{ij} = -\dfrac{\dot{\phi}}{\phi} \dot{w}_{ij} + \dfrac{c_i}{mc} \lambda_x^i x_j + \dfrac{k}{m} V_{\boldsymbol{\omega}_{ij}}(\boldsymbol{w}, t); \\ \dot{\lambda}_x^i = \left[-\dfrac{\dot{\phi}}{\phi} + \dfrac{d}{dt} \log(\sigma'(a_i)) + c_i \right] \lambda_x^i - \sigma'(a_i) \displaystyle\sum_{k \in \mathrm{ch}(i)} c_k \lambda_x^k w_{ki} - c L_{\xi_i}(x,t) \sigma'(a_i), \end{cases} \tag{16}$$

where λ_x^i is defined as in (15)*.*

Proof. See Appendix C. □

This in an interesting result especially since via the following corollary gives a direct link between the rescaled costates λ_x and the delta error of Backprop:

Corollary 1 (Reduction to Backprop)**.** *Let c_i be the same for all $i = 1, \dots, n$ so that now $c_i = c$, then the formal limit of the $\dot{\lambda}_x$ equation in the system 16 as $c \to \infty$ is*

$$\lambda_x^i = \sigma'(a_i) \sum_{k \in \mathrm{ch}(i)} \lambda_x^k w_{ki} + L_{\xi_i}(x,t) \sigma'(a_i). \tag{17}$$

Proof. Dividing both sides of the equation for λ_x^i in Eq. (16) by c we get:

$$\frac{\dot{\lambda}_x^i}{c} = \frac{1}{c} \left[-\frac{\dot{\phi}}{\phi} + \frac{d}{dt} \log(\sigma'(a_i)) \right] \lambda_x^i + \lambda_x^i - \sigma'(a_i) \sum_{k \in \mathrm{ch}(i)} \lambda_x^k w_{ki} - L_{\xi_i}(x,t) \sigma'(a_i).$$

As $c \to \infty$, the terms proportional to $1/c$ vanish, leaving us exactly with Eq. (17). □

Notice that Eq. (17) is exactly the update equation for delta errors in backpropagation: when i is an output neuron the value of λ is directly given by the gradient of the error, otherwise it is express as a sum on its children (see [13]).

4 From boundary to Cauchy's conditions

While discussing temporal locality in Section 3, we came across the problem of the left boundary conditions on the costate variables. We already noticed that these constraints spoil the locality of the differential equations that describe the minimality conditions of the variational problem at hand. In general, this prevents us from computing such solutions with a forward/causal scheme.

The following examples should suffice to explain that, in general, this is a crucial issue and should serve as motivation for the further investigation we propose in the present section.

[4] We have avoided to introduce the notation until now because we believe that it is worth writing (14) with the explicit dependence on the variable w and x at least once to better appreciate its overall structure.

https://doi.org/10.52202/079017-1989

Example 1. Consider a case in which $\ell(\boldsymbol{\omega}, \xi, s) \equiv V(\boldsymbol{\omega}, s)$, i.e. we want to study the minimization problem for $\int_0^T (m|\boldsymbol{v}(t)|^2/2 + V(\boldsymbol{w}(t;\boldsymbol{v}),t))\phi(t)dt$ under the constraint $\dot{\boldsymbol{w}} = \boldsymbol{v}$. Then the dynamical equation $\dot{x}(t) = f(x(t), \boldsymbol{w}(t))$ does not represent a constraint on variational problems for functional in (9). If we look at the Hamilton equation for $\dot{p}_x$ in (12) this reduces to $\dot{p}_x = -p_x \cdot f_{\boldsymbol{\omega}}$. We would however expect $p_x(t) \equiv 0$ for all $t \in [0,T]$. Indeed this is the solution that we would find if we pair $\dot{p}_x = -p_x \cdot f_{\boldsymbol{\omega}}$ with its boundary condition $p_x(T) = 0$ in (13). Notice that in general without this condition a random Cauchy initialization of this equation would not give null solution for the p_x. Now assume that $\phi = \exp(\theta t)$ with $\theta > 0$, and $m = 1$. Assume, furthermore[5] that $V(\boldsymbol{\omega}, s) = |\boldsymbol{\omega}|^2/2$. The functional $\int_0^T (|\dot{w}|/2 + |w|^2/2)e^{\theta t}dt$ defined over the functional space[6] $H^1([0,T];\mathbb{R}^N)$ is coercive and lower-semicontinuous, and hence admits a minimum (see [11]). Furthermore one can prove (see [20]) that such minimum is actually $C^\infty([0,T];\mathbb{R}^N)$. This allows us to describe such minumum with the Hamilton equations described in (12). In particular as we already commented the relevant equations are only that for $\dot{\boldsymbol{w}}$ and $\dot{p}_{\boldsymbol{w}}$ that is $\dot{\boldsymbol{w}}(t) = -p_{\boldsymbol{w}}(t)e^{-\theta t}$ and $\dot{p}_{\boldsymbol{w}}(t) = -\boldsymbol{w}e^{\theta t}$ with $p_{\boldsymbol{w}}(T) = 0$. This first order system of equations is equivalent to the second order differential equation $\ddot{\boldsymbol{w}}(t) + \theta\dot{\boldsymbol{w}}(t) - \boldsymbol{w}(t) = 0$. Each component of this second order system will, in general have an unstable behaviour since one of the eigenvalues is always real and positive. This is a strong indication that when solving Hamilton's equations with an initial condition on $p_{\boldsymbol{w}}$ we will end up with a solution that is far from the minimum.

In the next subsection, we will analyze this issue in more detail and present some alternative ideas that can be used to leverage Hamilton's equations for finding causal online solutions.

4.1 Time Reversal of the Costate

In Example 1 we discussed how the forward solution of Hamilton's (12) with initial conditions both on the state and on the costate in general cannot be related to any form of minimality of the cost function in (9) and this has to do with the fact that the proper minima are characterized also by left boundary conditions 13. The final conditions on $\dot{p}_x$ and $\dot{p}_{\boldsymbol{w}}$ suggest that the costate equations should be solved backward in time. Starting from the final temporal horizon and going backward in time is also the idea behind dynamic programming, which is of the main ideas at the very core of optimal control theory.

Autonomous systems of ODE with terminal boundary conditions can be solved "backwards" by time reversal [9, p. 597] operation $t \to -t$ and transforming terminal into initial conditions. More precisely the following classical result holds:

Propostion 3. *Let $\dot{y}(s) = \varphi(y(t))$ be a system of ODEs on $[0,T]$ with terminal conditions $y(T) = y^T$ and let ρ be the time reversal transformation maps $t \mapsto s = T - t$, then $\hat{y}(s) := y(\rho^{-1}(s)) = y(t)$ satisfies $\dot{\hat{y}}(s) = -\varphi(\hat{y}(s))$ with* initial *condition $\hat{y}(0) = y^T$.*

Clearly (12) or (16) are not an autonomous system and hence we cannot apply directly Proposition 3 nonetheless, we can still consider the following modification of (14)

$$\begin{cases} c_i^{-1}\dot{x}_i = -x_i + \sigma\Big(\sum_{j\in \mathrm{pa}(i)} w_{ij}x_j\Big) \\ \dot{w}_{ij} = -p^{ij}_{\boldsymbol{w}}/(mc\phi) \\ \dot{p}^i_x = -c_i p^i_x + \sum_{k\in \mathrm{ch}(i)} c_k \sigma'\Big(\sum_{j\in \mathrm{pa}(k)} w_{kj}x_j\Big)p^k_x w_{ki} + cL_{\xi_i}(x,t)\phi \\ \dot{p}^{ij}_{\boldsymbol{w}}(t) = c_i p^i_x \sigma'\Big(\sum_{m\in \mathrm{pa}(i)} w_{im}x_m\Big)x_j + ckV_{\boldsymbol{\omega}_{ij}}(\boldsymbol{w},t)\phi \end{cases} \tag{18}$$

which are obtained from (14) by changing the sign to $\dot{p}_x$ and $\dot{p}_{\boldsymbol{w}}$. Recalling the definition of the rescaled costates in (15) we can cast, in the same spirit of Proposition 2 a system of equations without $p_{\boldsymbol{w}}$. In particular we have as a corollary of Proposition 2 that

[5] The same argument that we give in this example works for a larger class of coercive potentials V.

[6] These are called Sobolev spaces, for more details see [5].

Corollary 2. *The ODE system in* (18) *is equivalent to*

$$\begin{cases} c_i^{-1}\dot{x}_i = -x_i + \sigma(a_i); \\ \ddot{w}_{ij} = -\dfrac{\dot{\phi}}{\phi}\dot{w}_{ij} - \dfrac{c_i}{mc}\lambda_x^i x_j - \dfrac{k}{m}V_{\boldsymbol{\omega}_{ij}}(\boldsymbol{w}, t); \\ \dot{\lambda}_x^i = \left[-\dfrac{\dot{\phi}}{\phi} + \dfrac{d}{dt}\log(\sigma'(a_i)) - c_i\right]\lambda_x^i + \sigma'(a_i)\displaystyle\sum_{k\in \mathrm{ch}(i)} c_k\lambda_x^k w_{ki} + cL_{\xi_i}(x,t)\sigma'(a_i), \end{cases} \tag{19}$$

Proof. Let us consider (16). The change of sign of $\dot{p}_{\boldsymbol{w}}$ only affect the signs of $\lambda_x^i x_j$ and $V_{\boldsymbol{\omega}_{ij}}(\boldsymbol{w}, t)$ in the $\ddot{w}_{ij}$ equation, while the change of sign of $\dot{p}_x$ result in a sign change of the term $c_i\lambda_x^i$, $\sigma'(a_i)\sum_{k\in \mathrm{ch}(i)} c_k\lambda_x^k w_{ki}$ and $L_{\xi_i}(x,t)\sigma'(a_i)$ in the equation for $\dot{\lambda}_i^x$. □

Equation 19 is indeed particularly interesting because it offers an interpretation of the dynamics of the weights w that is in the spirit of a gradient-base optimization method. In particular this allow us the extend the result that we gave in Corollary 1 to a full statement on the resulting optimization method:

Propostion 4 (GD with momentum)**.** *Let c_i be the same for all $i = 1, \dots, n$ so that now $c_i = c$, and let $\phi(t) = \exp(\theta t)$ with $\theta > 0$ then the formal limit of the system in* (19) *as $c \to \infty$ is*

$$\begin{cases} x_i = \sigma(a_i); \\ \ddot{w}_{ij} = -\theta\dot{w}_{ij} - \frac{1}{m}\lambda_x^i x_j - (k/m)V_{\boldsymbol{\omega}_{ij}}(\boldsymbol{w}, t); \\ \lambda_x^i = \sigma'(a_i)\sum_{k\in \mathrm{ch}(i)} \lambda_x^k w_{ki} + L_{\xi_i}(x,t)\sigma'(a_i). \end{cases} \tag{20}$$

Remark 2. This result shows that at least in the case of infinite speed of propagation of the signal across the network ($c \to \infty$) the dynamics of the weights prescribed by Hamilton's equation with the costate dynamics that is reversed (the sign of $\dot{p}_x$ and $\dot{p}_{\boldsymbol{w}}$ is changed) results in a gradient flow dynamic (heavy-ball dynamics) that it is interpretable as a gradient descent with momentum in the discrete. This is true since the term $\lambda_x^i x_j$ in this limit is exactly the Backprop factorization of the gradient of the term L with respect to the weights.

In view of this remark we can therefore conjecture that also for c fixed:

Conjecture 1. *Equation 19 is a* local optimization scheme *for the loss term ℓ.*

Such result would enable us to use (19) with initial Cauchy conditions as desired.

4.2 Continuous Time Reversal of State and Costate

Now we show that another possible approach to the problem of solving Hamilton's equation with Cauchy's conditions is to perform *simultaneous time-reversal* of both state and costate equation. Since in this case the sign flip involves both the Hamiltonian equations the approach is referred to as *Hamiltonian Sign Flip* (HSF). In order to introduce the idea let us begin with the following example.

Example 2 (LQ control)**.** Let us consider a linear quadratic scalar problem where the functional in (9) is $G(v) = \int_0^T qx^2/2 + rv^2/2\, dt$ and $\dot{x} = ax + bv$ with q, r positive and a and b real parameters. The associated Hamilton's equations in this case are

$$\dot{x} = ax - sp, \quad \dot{p} = -qx - ap, \tag{21}$$

where $s \equiv -b^2/r$. These equation can be solved with the ansatz $p(t) = \theta(t)x(t)$, where θ is some unknown parameter. Differentiating this expression with respect to time we obtain

$$\dot{\theta} = (\dot{p} - \theta\dot{x})/x, \tag{22}$$

and using the (21) into this expression we find $\dot{\theta} - s\theta^2 - 2a\theta - q = 0$ which is known as *Riccati equation*, and since $p(T) = 0$, because of boundary (13) this implies $\theta(T) = 0$. Again if instead we try to solve this equation with initial condition we end up with an unstable solution. However θ solves an autonomous ODE with final condition, hence by Proposition 3 we can solve it with 0 initial conditions as long as we change the sign of $\dot{\theta}$. Indeed the equation $\dot{\theta} + s\theta^2 + 2a\theta + q = 0$ is asymptotically stable and returns the correct solution of the Riccati algebraic equation. Now the crucial observation is that, as we can see from (22), the sign flip of $\dot{\theta}$ is equivalent to the *simultaneous* sign flip of $\dot{x}$ and $\dot{p}$.

https://doi.org/10.52202/079017-1989

In Example 2, as we observe from (22), the sign flip of $\dot{\theta}$ is equivalent to the *simultaneous* sign flip of $\dot{x}$ and $\dot{p}$. Inspired by the fact, let us associate the general Hamilton's equation ((12)), to this system the Cauchy problem

$$\begin{pmatrix} \dot{x}(t) \\ \dot{\boldsymbol{w}}(t) \\ \dot{p}_x(t) \\ \dot{p}_{\boldsymbol{w}}(t) \end{pmatrix} = s(t) \begin{pmatrix} f(x(t), \boldsymbol{w}(t), t) \\ -p_{\boldsymbol{w}}(t)/(mc\phi(t)) \\ -p_x(t) \cdot f_\xi(x(t), \boldsymbol{w}(t), t) - c\ell_\xi(\boldsymbol{w}(t), x(t), t)\phi(t) \\ -p_x(t) \cdot f_{\boldsymbol{\omega}}(x(t), \boldsymbol{w}(t), t) - c\ell_{\boldsymbol{\omega}}(\boldsymbol{w}(t), x(t), t)\phi(t) \end{pmatrix} \tag{23}$$

where for all $t \in [0, T]$, $s(t) \in \{0, 1\}$. Here we propose two different strategies that extends the sign flip discussed for the LQ problem.

Hamiltonian Track The basic idea is enforce system stabilization by choosing $s(t)$ to bound both the Hamiltonian variables. This leads to define a *Hamiltonian track*:

Definition 1. Let $S(\xi, \boldsymbol{\omega}, p, q) \subset (\mathbb{R}^{n-d} \times \mathbb{R}^N)^2$ for every $(\xi, \boldsymbol{\omega}, p, q) \in (\mathbb{R}^{n-d} \times \mathbb{R}^N)^2$ be a bounded connected set and let $t \mapsto X(t)$ any continuous trajectory in the space $(\mathbb{R}^{n-d} \times \mathbb{R}^N)^2$, then we refer to

$$\{(t, S(X(t)) : t \in [0, T]\} \in [0, T] \times (\mathbb{R}^{n-d} \times \mathbb{R}^N)^2$$

as *Hamiltonian track (HT)*.

Then we define $s(t)$ as follow

$$s(t) = \begin{cases} 1 & \text{if } (x(t), \boldsymbol{w}(t), p_x(t), p_{\boldsymbol{w}}(t)) \in S((x(t), \boldsymbol{w}(t), p_x(t), p_{\boldsymbol{w}}(t))) \\ -1 & \text{otherwise} \end{cases}. \tag{24}$$

For instance if we choose $S(\xi, \boldsymbol{\omega}, p, q) = \{(\xi, \boldsymbol{\omega}, p, q) : |\xi|^2 + |\boldsymbol{\omega}|^2 + |p|^2 + |q|^2 \leq R\}$ we are constraining the dynamics of (23) to be bounded since each time the trajectory $t \mapsto (x(t), \boldsymbol{w}(t), p_x(t), p_{\boldsymbol{w}}(t))$ moves outside of a ball of radius R we are reversing the dynamics by enforcing stability.

Hamiltonian Sign Flip Strategy and time reversal We can easily see that the sign flip driven by the policy of enforcing the system dynamics into the HT corresponds with time reversal of the trajectory, which can nicely be interpreted as focus of attention mechanism. A simple approximation of the movement into the HT is that of selecting $s(t) = \text{sign}(\cos(\bar{\omega}t))$, where $\bar{\omega} = 2\pi\bar{f}$ is an appropriate *flipping frequency* which governs the movement into the HT. In the discrete setting of computation the strategy consists of flipping the right-side of Hamiltonian equations sign with a given period. In the extreme case the sign flip takes place at any Euler discretization step. Here we report the application of the *Hamiltonian Sign Flip* strategy to the classic Linear Quadratic Tracking (LQT) problem by using a recurrent neural network based on a fully-connected digraph. The purpose of the reported experiments is to validate the HSF policy, which is in fact of crucial importance in order to exploit the power of the local propagation presented in the paper, since the proposed policy enables on-line processing.

The pre-algorithmic framework proposed in the paper, which is based on ODE can promptly give rise to algorithmic interpretations by numerical solutions. In the reported experiments we used Euler's discretization (see Appendix E for both architectural and algorithmic details).

Sinusoidal signals: The effect of the accuracy parameter. In this experiment we used a sinusoidal target and a recurrent neural network with five neurons, while the objective function was $G(v) = \int_0^T q(x_0 - z)^2/2 + r|v|^2/2 + r_w|w|^2\, dt$, where we also introduced a regularization term on the weights. Here, x_0 denotes the neuron designated as the output (see Appendix E) and q, r and r_w are positive parameters. The HSF policy gives rise to the expected approximation results. In Fig. 1–2 we can appreciate the effect of the increment of the accuracy term.

Tracking under hard predictability conditions. This experiment was conceived to assess the capabilities of the same small recurrent neural network with five neurons to track a signal which was purposely generated to be quite hard to predict. It is composed of patching intervals with cosine functions with constants.

The experimental analysis on this and related examples confirms effectiveness of the HSF policy shown in Fig. 3. Figure 4 shows the behavior of the Lagrangian and of the Hamiltonian term, with the latter providing insights into the energy exchange with the environment.

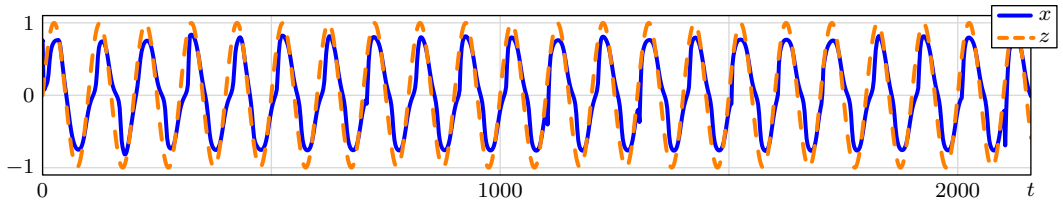

Figure 1: Recurrent net with 5 neurons, $q = 10$ (accuracy term), $r_w = 1$ (weight regularization term), $r = 0.1$ (derivative of the weight term).

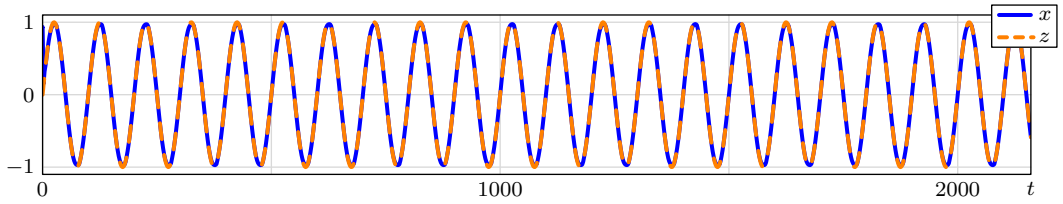

Figure 2: Recurrent net with 5 neurons, $q = 1000$ (accuracy term), $r_w = 1$ (weight regularization term), $r = 0.1$ (derivative of the weight term).

5 Related Work

Optimal control. Optimal control theory primarily studies minimality problems for dynamical systems [1, 6]. The two main complementary approaches to the problem are the Pontryagin Maximum Principle [10] and dynamic programming. Additionally, as a general minimization problem, both approaches significantly intersect with the calculus of variations [12]. Optimal control for discrete problems is also a classic topic [2, p. 2].

Neural ODE. Recent works, such as [7] and subsequent studies [17, 24] have applied results from optimal control to develop learning algorithms based on differential equations. However, these approaches differ significantly from the continual online learning considered in this work, as the time variable in the class of ODEs they examine is not tied to the input signal that represents the flow of the learning environment.

Online. On the other hand several works propose to formulate the learning problems online and from a single stream of data [22, 34]. The classical approach to learn RNNs online is RTRL [15]; several approaches has been since proposed to reduce the high space/time complexities due to the progressive update of a Jacobian matrix [23]. In our method no storing of Jacobian matrices happens, hence the proposed method is not a generalization/reformulation of RTRL not related approaches like [35].

Nature-inspired computations. The primary distinction of our approach in discussing the biological plausibility of backpropagation lies in our development of a theory grounded entirely in temporal analysis within the environment and the concept of learning over time. While several classical [28] and recent approaches [29, 25, 26, 27, 14] share certain locality properties outlined here, they are primarily inspired by brain physiology. Similarly, most works that examine the biological plausibility of backpropagation [8, 33, 21] overlook the role of time in the sense that we present in this work. Here, we propose laws of neural propagation where connections are updated progressively over time, mirroring processes observed in nature.

6 Conclusions

This paper is motivated by the idea of a proposing learning scheme that, like in nature, arises without needing data collections, but simply by on-line processing of the environmental interactions. The paper gives two main contributions. First, it introduces a local spatiotemporal pre-algorithmic framework that is inspired to classic Hamiltonian equations. It is shown that the corresponding algorithmic formalization leads to the interpretation of Backpropagation as a limit case of the

https://doi.org/10.52202/079017-1989

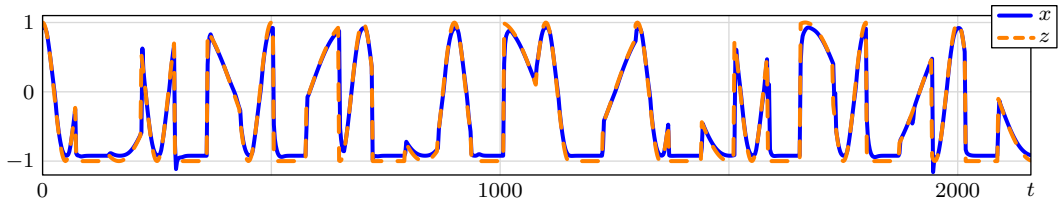

Figure 3: Tracking a highly-unpredictable signal: number of neurons: 5, $q = 100$ (accuracy), weight reg = 1, derivative of weight reg = 0.1.

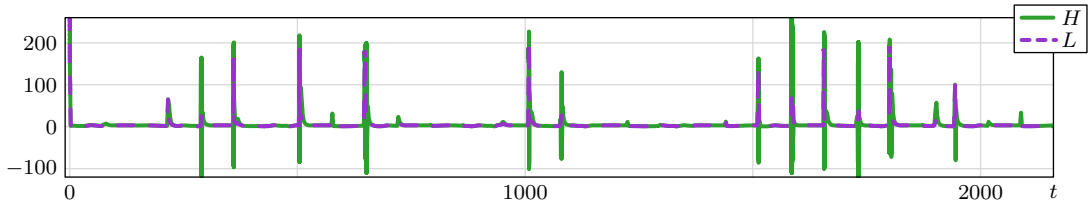

Figure 4: Evolution of the Lagrangian and of the Hamiltonian function for the experiment whose tracking is shown in Fig. 3.

proposed diffusion process in case of infinite velocity. This sheds light on the longstanding discussion on the biological plausibility of Backpropagation, since the proposed computational scheme is local in both space and time. This strong result is indissolubly intertwined with a strong limitation. The theory enables such a locality under the assumption that the associated ordinary differential equations are solved as a boundary problem. The second result of the paper is that of proposing a method for approximating the solution of the Hamiltonian problem with boundary conditions by using Cauchy's initial conditions. In particular we show that we can stabilize the learning process by appropriate schemes of time reversal that are related to focus of attention mechanisms. We provide experimental evidence of the effect of the proposed Hamiltonian Sign Flip policy for problems of tracking in automatic control. While the proposed local propagation scheme is optimal in the temporal setting and overcomes the limitations of classic related learning algorithms like BPTT and RTRL, the given results show that there is no free lunch: The distinguishing feature of spatiotemporal locality needs to be sustained by appropriate movement policies into the Hamiltonian Track. We expect that other solutions better than the HSF policy herein proposed can be developed when dealing with real-word problems and may offer potential approaches to classic challenges in lifelong learning, such as forgetting, that remain open and are not fully addressed by the current framework. This paper must only be regarded as a theoretical contribution which offers a new pre-algorithmic view of neural propagation. While the provided experiments support the theory, the application to real-world problems need to activate substantial joint research efforts on different application domains.

Acknowledgments

We thank Stefano Melacci and Giovanni Bellettini for insightful discussions.

References

[1] Martino Bardi, Italo Capuzzo Dolcetta, et al. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*, volume 12. Springer, 1997.

[2] Dimitri Bertsekas. *Abstract dynamic programming*. Athena Scientific, 2022.

[3] Alessandro Betti, Marco Gori, and Stefano Melacci. Cognitive action laws: The case of visual features. *IEEE transactions on neural networks and learning systems*, 31(3):938–949, 2019.

[4] Alessandro Betti, Marco Gori, and Stefano Melacci. Learning visual features under motion invariance. *Neural Networks*, 126:275–299, 2020.

[5] Haim Brezis and Haim Brézis. *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer, 2011.

[6] Piermarco Cannarsa and Carlo Sinestrari. *Semiconcave functions, Hamilton-Jacobi equations, and optimal control*, volume 58. Springer Science & Business Media, 2004.

[7] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[8] Francis Crick. The recent excitement about neural networks. *Nature*, 337(6203):129–132, 1989.

[9] Lawrence C Evans. *Partial differential equations*, volume 19. American Mathematical Society, 2010.

[10] RV Gamkrelidze, Lev Semenovich Pontrjagin, and Vladimir Grigor'evic Boltjanskij. *The mathematical theory of optimal processes*. Macmillan Company, 1964.

[11] Mariano Giaquinta and Stefan Hildebrandt. Calculus of variations i. *Calculus of Variations*, 1:1, 1995.

[12] Mariano Giaquinta and Stefan Hildebrandt. *Calculus of variations II*, volume 311. Springer Science & Business Media, 2013.

[13] Marco Gori, Alessandro Betti, and Stefano Melacci. *Machine Learning: A constraint-based approach*. Elsevier, 2023.

[14] Geoffrey Hinton. The forward-forward algorithm: Some preliminary investigations. *arXiv preprint arXiv:2212.13345*, 2022.

[15] Kazuki Irie, Anand Gopalakrishnan, and Jürgen Schmidhuber. Exploring the promise and limits of real-time recurrent learning. *arXiv preprint arXiv:2305.19044*, 2023.

[16] Jack Kendall. A gradient estimator for time-varying electrical networks with non-linear dissipation. *arXiv preprint arXiv:2103.05636*, 2021.

[17] Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. *Advances in Neural Information Processing Systems*, 33:6696–6707, 2020.

[18] Axel Laborieux and Friedemann Zenke. Holomorphic equilibrium propagation computes exact gradients through finite size oscillations. *Advances in Neural Information Processing Systems*, 35:12950–12963, 2022.

[19] Yann LeCun, D Touresky, G Hinton, and T Sejnowski. A theoretical framework for backpropagation. In *Proceedings of the 1988 connectionist models summer school*, volume 1, pages 21–28, 1988.

[20] Matthias Liero and Ulisse Stefanelli. A new minimum principle for lagrangian mechanics. *Journal of nonlinear science*, 23:179–204, 2013.

[21] Timothy P Lillicrap, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6):335–346, 2020.

[22] Zheda Mai, Ruiwen Li, Jihwan Jeong, David Quispe, Hyunwoo Kim, and Scott Sanner. Online continual learning in image classification: An empirical survey. *Neurocomputing*, 469:28–51, 2022.

[23] Owen Marschall, Kyunghyun Cho, and Cristina Savin. A unified framework of online learning algorithms for training recurrent neural networks. *Journal of machine learning research*, 21(135):1–34, 2020.

[24] Stefano Massaroli, Michael Poli, Jinkyoo Park, Atsushi Yamashita, and Hajime Asama. Dissecting neural odes. *Advances in Neural Information Processing Systems*, 33:3952–3963, 2020.

[25] Beren Millidge, Alexander Tschantz, and Christopher L Buckley. Predictive coding approximates backprop along arbitrary computation graphs. *Neural Computation*, 34(6):1329–1368, 2022.

 https://doi.org/10.52202/079017-1989

[26] Alexander Ororbia and Ankur Mali. The predictive forward-forward algorithm. *arXiv preprint arXiv:2301.01452*, 2023.

[27] Alexander G Ororbia and Ankur Mali. Biologically motivated algorithms for propagating local target representations. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pages 4651–4658, 2019.

[28] R Rao. Predictive coding in the visual cortex. *Nature Neuroscience*, 2(1):9–10, 1999.

[29] Tommaso Salvatori, Ankur Mali, Christopher L Buckley, Thomas Lukasiewicz, Rajesh PN Rao, Karl Friston, and Alexander Ororbia. Brain-inspired computational intelligence via predictive coding. *arXiv preprint arXiv:2308.07870*, 2023.

[30] Benjamin Scellier. A deep learning theory for neural networks grounded in physics. *arXiv preprint arXiv:2103.09985*, 2021.

[31] Benjamin Scellier and Yoshua Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24, 2017.

[32] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural networks. *Physical review letters*, 61(3):259, 1988.

[33] David Stork. Is backpropagation biologically plausible? In *International 1989 Joint Conference on Neural Networks*, pages 241–246. IEEE, 1989.

[34] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[35] Nicolas Zucchet, Robert Meier, Simon Schug, Asier Mujika, and João Sacramento. Online learning of long-range dependencies. *Advances in Neural Information Processing Systems*, 36:10477–10493, 2023.

A Optimal Control

The classical way in which Hamilton's equations are derived is through Hamilton-Jacobi-Bellman theorem. So let enunciate this theorem in a general setting. Here we use the notation $y = (x, \boldsymbol{w})$ to stand for the whole state vector and $p = (p_x, p_{\boldsymbol{w}})$. We will also denote with α the control parameters. Moreover to avoid cumbersome notation in this appendix we will override the notation on the symbols n and N and we will use them here to denote the dimension of the state and of the control parameters respectively.

A.1 Hamilton Jacobi Bellman Theorem

Consider the classical state model

$$\dot{y}(t) = f(y(t), \alpha(t), t), \quad t \in (t_0, T] \tag{25}$$

$f\colon \mathbb{R}^n \times \mathbb{R}^N \times [t_0, T] \to \mathbb{R}^n$ is a Lipschitz function, $t \mapsto \alpha(t)$ is the trajectory of the parameters of the model, which is assumed to be a *measurable function* with assigned initial state $y^0 \in \mathbb{R}^n$, that is

$$y(t_0) = y^0. \tag{26}$$

Let us now pose $\mathcal{A} := \{\alpha\colon [t_0, T] \to \mathbb{R}^N : \alpha \text{ is measurable}\}$ and given a $\beta \in \mathcal{A}$, and given an initial state y^0, we define the *state trajectory*, that we indicate with $t \mapsto x(t; \beta, y^0, t_0)$, the solution of (25) with initial condition (26).

Now let us define a cost functional C that we want to minimize:

$$C_{y^0,t_0}(\alpha) := \int_{t_0}^{T} \Lambda(\alpha(t), y(t; \alpha, y^0, t_0), t)\, dt, \tag{27}$$

where $\Lambda(a, \cdot, s)$ is bounded and Lipshitz $\forall a \in \mathbb{R}^N$ and $\forall s \in [t_0, T]$. Then the problem

$$\min_{\alpha \in \mathcal{A}} C_{y^0,t_0}(\alpha) \tag{28}$$

is a constrained minimization problem which is usually denoted as *control problem* [1], assuming that a solution exists. The first step to address our constrained minimization problem is to define the *value function* or *cost to go*, that is a map $v\colon \mathbb{R}^n \times [t_0, T] \to \mathbb{R}$ defined as

$$v(\xi, s) := \inf_{\alpha \in \mathcal{A}} C_{\xi,s}(\alpha), \quad \forall (\xi, s) \in \mathbb{R}^n \times [t_0, T]$$

and the Hamiltonian function $H\colon \mathbb{R}^n \times \mathbb{R}^n \times [t_0, T] \to \mathbb{R}$ as

$$H(\xi, \rho, s) := \min_{a \in \mathbb{R}^N} \{\rho \cdot f(\xi, a, s) + \Lambda(a, \xi, s)\}, \tag{29}$$

being $\cdot$ the dot product. Then Hamilton-Jacobi-Bellman theorem states that

Theorem 3 (Hamilton-Jacobi-Bellman)**.** *Let us assume that D denotes the gradient operator with respect to ξ. Furthermore, let us assume that $v \in C^1(\mathbb{R}^n \times [t_0, T], \mathbb{R})$ and that the minimum of $C_{\xi,s}$, Eq.* (28)*, exists for every $\xi \in \mathbb{R}^n$ and for every $s \in [t_0, T]$. Then v solves the PDE*

$$v_s(\xi, s) + H(\xi, Dv(\xi, s), s) = 0, \tag{30}$$

$(\xi, s) \in \mathbb{R}^n \times [t_0, T)$, with terminal condition $v(\xi, T) = 0$, $\forall \xi \in \mathbb{R}^n$. Equation 30 is usually referred to as Hamilton-Jacobi-Bellman equation.

Proof. Let $s \in [t_0, T)$ and $\xi \in \mathbb{R}^n$. Furthermore, instead of the optimal control let us use a constant control $\alpha_1(t) = a \in \mathbb{R}^N$ for times $t \in [s, s + \epsilon]$ and then the optimal control for the remaining temporal interval. More precisely let us pose

$$\alpha_2 \in \arg\min_{\alpha \in \mathcal{A}} C_{y(s+\varepsilon; a, \xi, s), s+\varepsilon}(\alpha).$$

Now consider the following control

$$\alpha_3(t) = \begin{cases} \alpha_1(t) & \text{if } t \in [s, s + \varepsilon) \\ \alpha_2(t) & \text{if } t \in [s + \varepsilon, T]\,. \end{cases} \tag{31}$$

https://doi.org/10.52202/079017-1989

Then the cost associated to this control is

$$\begin{aligned} C_{\xi,s}(\alpha_3) &= \int_s^{s+\varepsilon} \Lambda(a, y(t;a,\xi,s), t)\, dt \\ &\quad + \int_{s+\varepsilon}^{T} \Lambda(\alpha_2(t), y(t;\alpha_2,\xi,s), t)\, ds \\ &= \int_s^{s+\varepsilon} \Lambda(a, y(t;a,\xi,s), t)\, dt \\ &\quad + v(y(s+\varepsilon;a,\xi,s), s+\varepsilon) \end{aligned} \tag{32}$$

By definition of value function we also have that $v(\xi,s) \leq C_{\xi,s}(\alpha_3)$. When rearranging this inequality, dividing by ε, and making use of the above relation we have

$$\begin{aligned} &\frac{v(y(s+\varepsilon;a,\xi,s), s+\varepsilon) - v(\xi,s)}{\varepsilon} + \\ &\frac{1}{\varepsilon}\int_s^{s+\varepsilon} \Lambda(a, y(t;a,\xi,s), t)\, dt \geq 0 \end{aligned} \tag{33}$$

Now taking the limit as $\varepsilon \to 0$ and making use of the fact that $y'(s,a,\xi,s) = f(\xi,a,s)$ we get

$$v_s(\xi,s) + Dv(\xi,s)\cdot f(\xi,a,s) + \Lambda(a,\xi,s) \geq 0. \tag{34}$$

Since this inequality holds for any chosen $a \in \mathbb{R}^N$ we can say that

$$\inf_{a\in\mathbb{R}^N} \{v_s(\xi,s) + Dv(\xi,s)\cdot f(\xi,a,s) + \Lambda(a,\xi,s)\} \geq 0 \tag{35}$$

Now we show that the $\inf$ is actually a $\min$ and, moreover, that minimum is 0. To do this we simply choose $\alpha^* \in \arg\min_{\alpha\in\mathcal{A}} C_{\xi,s}(\alpha)$ and denote $a^* := \alpha^*(s)$, then

$$\begin{aligned} v(\xi,s) &= \int_s^{s+\varepsilon} \Lambda(\alpha^*(t), y(t;\alpha^*,\xi,s), t)\, dt \\ &\quad + v(y(s+\varepsilon;\alpha^*,\xi,s). \end{aligned} \tag{36}$$

Then again dividing by ε and using that $y'(s;\alpha^*,\xi,a) = f(\xi,a^*,s)$ we finally get

$$v_s(\xi,s) + Dv(\xi,s)\cdot f(\xi,a^*,s) + \Lambda(a^*,\xi,s) = 0 \tag{37}$$

But since $a^* \in \mathbb{R}^N$ and we knew that $\inf_{a\in\mathbb{R}^N}\{v_s(\xi,s) + Dv(\xi,s)\cdot f(\xi,a,s) + \Lambda(a,\xi,s)\} \geq 0$ it means that

$$\begin{aligned} &\inf_{a\in\mathbb{R}^N} \{v_s(\xi,s) + Dv(\xi,s)\cdot f(\xi,a,s) + \Lambda(a,\xi,s)\} = \\ &\min_{a\in\mathbb{R}^N} \{v_s(a,s) + Dv(\xi,s)\cdot f(\xi,a,s) + \Lambda(a,\xi,s\} = 0. \end{aligned} \tag{38}$$

Recalling the definition of H we immediately see that the last inequality is exactly (HJB). □

A.2 Hamilton Equations: The Method of Characteristics

Now let us define $p(t) = Dv(y(t),t)$ so that by definition of the value function $p(T) = 0$ which gives (13). Also by differentiating this expression with respect to time we have

$$\dot{p}_k(t) = v_{\xi_k t}(y(t),t) + \sum_{i=1}^{n} v_{\xi_k\xi_i}(y(t),t)\cdot \dot{y}_i. \tag{39}$$

Now since v solves (30), if we differentiate the Hamilton Jacobi equation by ξ_k we obtain:

$$v_{t\xi_k}(\xi,s) = -H_{\xi_k}(\xi, Dv(\xi,s), s) - \sum_{i=1}^{n} H_{\rho_i}(\xi, Dv(\xi,s), s)\cdot v_{\xi_k\xi_i}(\xi,s).$$

Once we compute this expression on $(y(t),t)$ and we substitute it back into (39) we get:

$$\dot{p}_k(t) = -H_{\xi_k}(y(t), Dv(y(t),t),t) + \sum_{i=1}^{n}\Big[\dot{y}_i(t) - H_{\rho_i}(y(t), Dv(y(t),t),t)\Big]\cdot v_{\xi_k\xi_i}(y(t),t).$$

Now if we choose $y$so that it satisfies $\dot{y}(t) = H_\rho(y(t),p(t),t)$ the above equation reduces to

$$\dot{p} = -H_\xi(y(t),p(t),t).$$

Applying these equations to the Hamiltonian in (11) we indeed end up with (12).

B Proof of Theorem 2

From (7) and the hypothesis on ℓ we have that

$$f^k_{\xi_i} = -c_i\delta_{ik} + c_k\sigma'\Big(\sum_{j\in\mathrm{pa}(k)} w_{kj}x_j\Big)\sum_{m\in\mathrm{pa}(k)} w_{km}\delta_{mi}, \qquad \ell_\xi = L_{\xi_i}(x,t)$$
$$f^k_{\boldsymbol{\omega}_{ij}} = c_k\sigma'\Big(\sum_{m\in\mathrm{pa}(k)} w_{km}x_m\Big)\sum_{h\in\mathrm{pa}(k)} \delta_{ik}\delta_{jh}x_h, \qquad \ell_{\boldsymbol{\omega}_{ij}} = kV_{\boldsymbol{\omega}_{ij}}.$$

Then (12) becomes

$$\begin{cases} c_i^{-1}\dot{x}_i = -x_i + \sigma\Big(\sum_{j\in\mathrm{pa}(i)} w_{ij}x_j\Big) \\ \dot{w}_{ij} = -p^{ij}_{\boldsymbol{w}}/(mc\phi) \\ \dot{p}^i_x = c_ip^i_x - \sum_{k=d+1}^n \sum_{m\in\mathrm{pa}(k)} c_kp^k_x\sigma'\Big(\sum_{j\in\mathrm{pa}(k)} w_{kj}x_j\Big)w_{km}\delta_{mi} - cL_{\xi_i}(x,t)\phi \\ \dot{p}^{ij}_{\boldsymbol{w}}(t) = -\sum_{k=d+1}^n c_kp^k_x\sigma'\Big(\sum_{m\in\mathrm{pa}(k)} w_{km}x_m\Big)\sum_{h\in\mathrm{pa}(k)} \delta_{ik}\delta_{jh}x_h - ckV_{\boldsymbol{\omega}_{ij}}(\boldsymbol{w},t)\phi \end{cases} \tag{40}$$

Now to conclude the proof it is sufficient to apply the following lemma to conveniently rewrite and switch the sums in the $\dot{p}$ equations.

Lemma 1. *Let A be the set of the arches of a digraph as in Section 2, and let* (2) *be true, then*

$$A = \{\,(m,k)\in A : k\in\{d+1,\dots,n\}\,\} = \{\,(m,k)\in A : m\in\{1,\dots,n\}\,\}.$$

Equivalently we may say that $\sum_{k=d+1}^n\sum_{m\in\mathrm{pa}(k)} = \sum_{m=1}^n\sum_{k\in\mathrm{ch}(m)}$.

Proof. It is an immediate consequences of the fact that the first d neurons are all parents of some neuron in $\{d+1,\dots,n\}$ ((2)) and that they do not have themselves any parents ((1)). □

C Proof of Proposition 2

The first equation of (16) is a simple rewriting of the first expression in (14), utilizing the definition of activation from Eq. (15). We obtain the second equation in (16) by combining the second and last equations in Eq .(14):

$$\ddot{w}_{ij} = -\dot{p}^{ij}_{\boldsymbol{w}}/(mc\phi) + p^{ij}_{\boldsymbol{w}}\dot{\phi}/(mc\phi^2).$$

The expression for $\dot{p}^{ij}_{\boldsymbol{w}}$ can be substituted from the last equation in Eq. (14), with $p^{ij}_{\boldsymbol{w}} = -mc\phi\dot{w}_{ij}$. Finally, $p^i_x = \phi\sigma'(a_i)\lambda^i_x$ from Eq. (15). To derive the second equation in (16), we start by differentiating λ^i_x as defined in Eq. (15), obtaining:

$$\dot{\lambda}^i_x = \frac{\sigma''(a_i)}{\phi}p^i_x - \sigma'(a_i)\frac{\dot{\phi}}{\phi^2}p^i_x + \frac{\sigma'(a_i)}{\phi}\dot{p}^i_x.$$

We then substitute $p^i_x = \phi\sigma'(a_i)\lambda^i_x$ as above and use the third equation in (14) for $\dot{p}^i_x$. This equation, with all p_x terms converted to λ_x as per $p^i_x = \phi\sigma'(a_i)\lambda^i_x$, yields the exact expression for λ^i_x in Eq. (16).

D Proof of Proposition 1

Let $\mu(t) := \sigma\Big(\sum_{j\in\mathrm{pa}(i)} w_{ij}x_j(t)\Big)$ be. From the boundedness of $\sigma(\cdot)$ we know that there exists $B>0$ such that $|\mu(t)|\le B$. Now we have

$$\begin{aligned} x_i(t) = x_i(0)e^{-\alpha t} + \int_0^t e^{-\alpha(t-\tau)}u(\tau)d\tau &\le x_i(0) + B\int_0^t e^{-\alpha(t-\tau)}d\tau \\ &\le x_i(0) + \frac{B}{\alpha}(1-e^{-t}) < x_i(0) + \frac{B}{\alpha} \end{aligned}$$

https://doi.org/10.52202/079017-1989

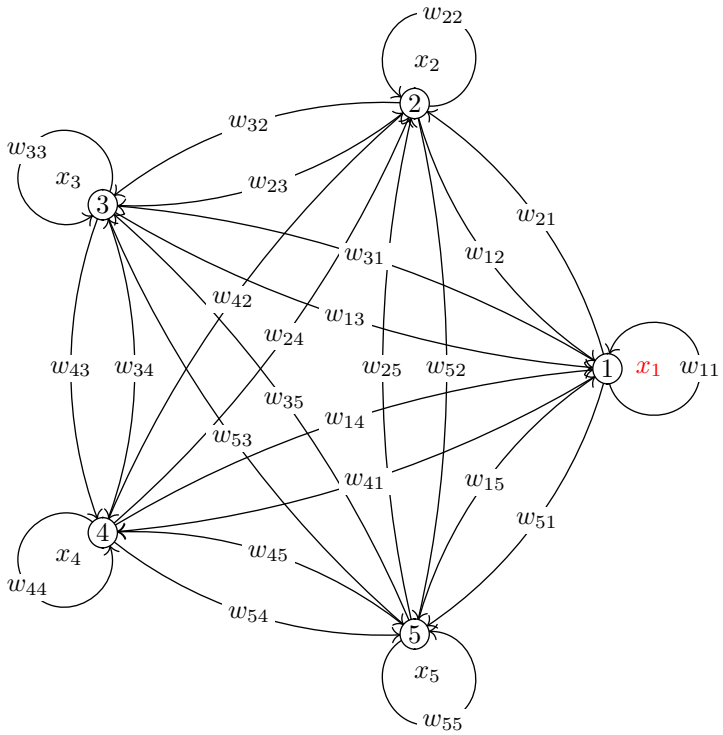

Figure 5: Architecture used in the experiments. The red neuron is the one that is used as output and it is forced to follow the reference (target) signal.

E Architectural and Algorithmic details

Figure 5 illustrates the network architecture used in the experiments described in in Section 4.2. Algorithm 1 provides a detailed explanation of the Hamiltonian Sign Flip method, which is also discussed in the same section and applied in all the experimental results presented.

Algorithm 1 Hamiltonian Sign Flip. In red the change of signs due to HSF. The locality of the method is evident from the loop on time t while the spatial locality depends on the structure of each update rule for the states and costates. (what we propose is valid also for unevenly spaced data).

Init $x^0 = \text{rand}$, $\boldsymbol{w}^0 = \text{rand}$, $p_x^0 = 0$, $p_{\boldsymbol{w}}^0 = 0$. Select $c > 0$ and choose function ϕ.
while $t < T$ **do**
  Compute s^t using Eq. (24).
  $\dot{x}^t \leftarrow f(x^t, \boldsymbol{w}^t, t)$, $\quad \dot{x}^t \leftarrow s^t \dot{x}^t$
  $\dot{\boldsymbol{w}}^t \leftarrow -p_{\boldsymbol{w}}^t/(mc\phi^t)$, $\quad \dot{\boldsymbol{w}}^t \leftarrow s^t \dot{\boldsymbol{w}}^t$
  $\dot{p}_x^t \leftarrow -p_x^t \cdot f_{\xi}(x^t, \boldsymbol{w}^t, t) - c\ell_{\xi}(\boldsymbol{w}^t, x^t, t)\phi^t$, $\quad \dot{p}_x^t \leftarrow s^t \dot{p}_x^t$
  $\dot{p}_{\boldsymbol{w}}^t \leftarrow -p_x^t \cdot f_{\boldsymbol{u}}(x^t, \boldsymbol{w}^t, t) - c\ell_{\boldsymbol{u}}(\boldsymbol{w}^t, x^t, t)\phi^t$, $\quad \dot{p}_{\boldsymbol{w}}^t \leftarrow s^t \dot{p}_{\boldsymbol{w}}^t$
  $x^{t+\tau} = x^t + \tau \dot{x}^t$
  $\boldsymbol{w}^{t+\tau} = \boldsymbol{w}^t + \tau \dot{\boldsymbol{w}}^t$
  $p_x^{t+\tau} = p_x^t + \tau \dot{p}_x^t$
  $p_{\boldsymbol{w}}^{t+\tau} = p_{\boldsymbol{w}}^t + \tau \dot{p}_{\boldsymbol{w}}^t$
  $t = t + \tau$
end while

https://doi.org/10.52202/079017-1989

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: -

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The main limitation of this paper are described in the Conclusion section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

https://doi.org/10.52202/079017-1989

Justification: All claims are properly stated and proofs are provided either in the main paper or in the appendices.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The code with instructions to reproduce the experiments is provided.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code with instructions to reproduce the experiments is provided, no external data is needed to reproduce the experiments.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The experiments only assess a qualitative behaviour of a newly introduced learning rules.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

https://doi.org/10.52202/079017-1989

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Computing requirements of the experiment are so modest that any modern laptop can sustain it.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have read the NeurIPS Code of Ethics and in our opinion we are not in violation of any norm therein contained.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no direct clear foreseeable either positive or negative social impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: -

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: -

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

https://doi.org/10.52202/079017-1989

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: -

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: -

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: -

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.