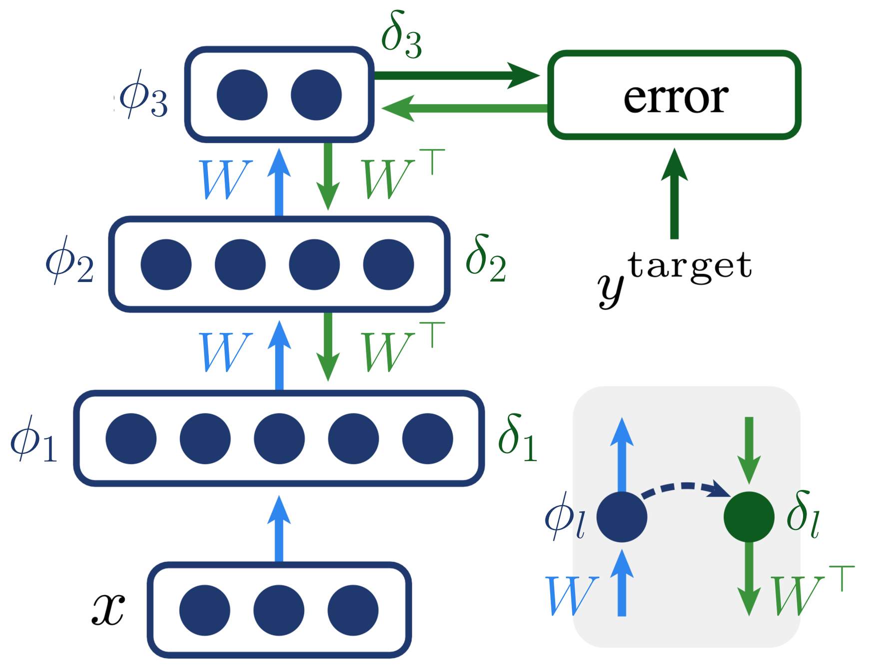
The least-control principle for local learning at equilibrium

Alexander Meulemans*, Nicolas Zucchet*, Seijin Kobayashi*, Johannes von Oswald, João Sacramento

ETH zürich

Selected as an oral — Neural Information Processing Systems

Backpropagation in the brain?



Issues:
- weight transport
- clocked two-phased algorithm
- activity not changed
- no recurrence allowed
- differentiability

state update	$\phi_{l+1} = W_{l+1}\rho(\phi_l)$
error update	$\delta_l = \rho'(\phi_l)W_{l+1}^\top \delta_{l+1}$
weight update	$\Delta W_{l+1} \propto \delta_{l+1}\rho(\phi_l)^\top$

Grossberg, *Cogn. Sci.* 1987; Crick, *Nature* 1989; Lillicrap et al., *Nat. Rev. Neurosci.* 2020

Desiderata for our theory

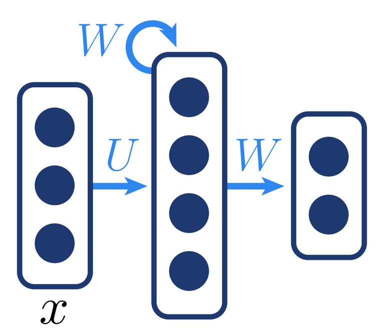
Evidence from biology that we account for
1. local and activity-dependent learning rules
2. the brain is recurrently connected

Reasonable assumptions for learning in the brain
1. gradient-based learning
2. single-phase learning
3. feedback modulation to embed credit assignment information in the neural activity

Equilibrium neural networks

From feedforward networks to equilibrium networks:

$\phi_{l+1} = W_{l+1}\rho(\phi_l)$	deep learning description
$\dot\phi_{l+1} = -\phi_{l+1} + W_{l+1}\rho(\phi_l)$	leaky-integrate and fire until equilibrium
$\dot\phi = -\phi + W\rho(\phi) + Ux$	more general connectivity patterns (possibly recurrent)

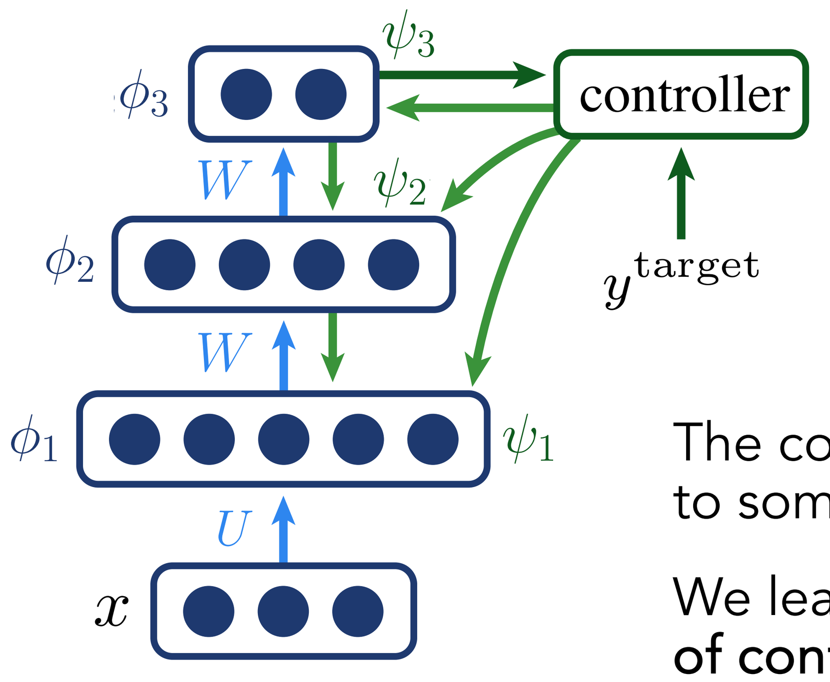


recursion is a powerful algorithmic primitive

such recurrent models can be more memory efficient

Bai et al., *NeurIPS.* 2019

Controlling neural activity



objective of the controller
$\phi_y = y^{\text{target}}$

controlled dynamics
$\dot\phi = -\phi + W\rho(\phi) + Ux + \psi$

The controller pushes neural activity to some target value

We learn by **minimizing the amount of control** needed at equilibrium

Sussillo & Abbott, *Neuron* 2009; Gilra & Gerstner, *eLife* 2017; Alemi et al., *AAAI* 2018; Podlaski & Machens, *NeurIPS* 2020

The least-control principle

$$\min_{\phi,\psi,\theta} \frac{1}{2}\|\psi\|^2 \quad \text{minimize amount of control}$$

$$\text{s.t.} \quad 0 = \phi + W\rho(\phi) + Ux + \psi \quad \text{controlled equilibrium}$$

$$\text{and} \quad \phi_y = y^{\text{target}} \quad \text{correct output}$$

Step 1: find an optimal control ψ_* and an optimally controlled state ϕ_*
How to compute it? see right panel

Step 2: take a gradient descent step in the least-control objective $\|\psi_*\|^2/2$ w.r.t. the parameters θ

$$\Delta W \propto \psi_* \rho(\phi_*)^\top$$

single-phased, local, activity-dependent, gradient-based!

If the model has enough capacity, the update will decrease the amount of optimal to zero and solve the learning task

The theory also applies to any system reaching an equilibrium (e.g. meta-learning)

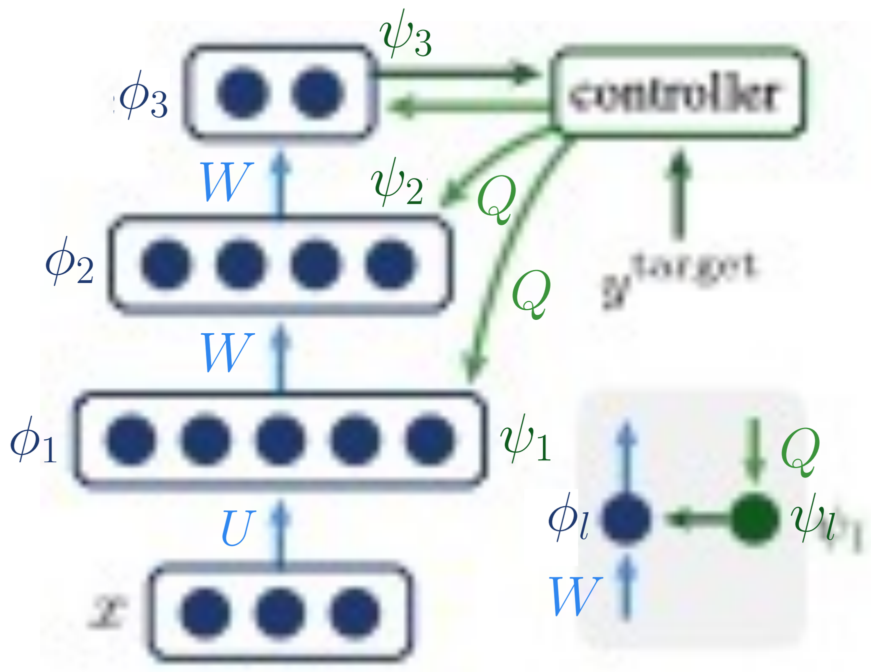
Connection to existing theories

augmented energy: $\frac{1}{2}\|-\phi + W\rho(\phi) + Ux\|^2 + \frac{\beta}{2}\|\phi_y - y^{\text{target}}\|^2$

$\beta \to 0$	β	$\beta \to \infty$
weak nudging	⟷	*perfect control*
equilibrium propagation		predictive coding least-control principle

Scellier & Bengio, *Front. Comput. Neurosci.* 2017; Rao & Ballard, *Nat. Neurosci.* 1999; Friston, *Trends Cogn. Sci.* 2009; Whittington & Bogacz, *Neur. Comp.* 2017

Direct linear feedback



output controller that forces the output to be at target value
$\dot u = -(\phi_y - y^{\text{target}})$

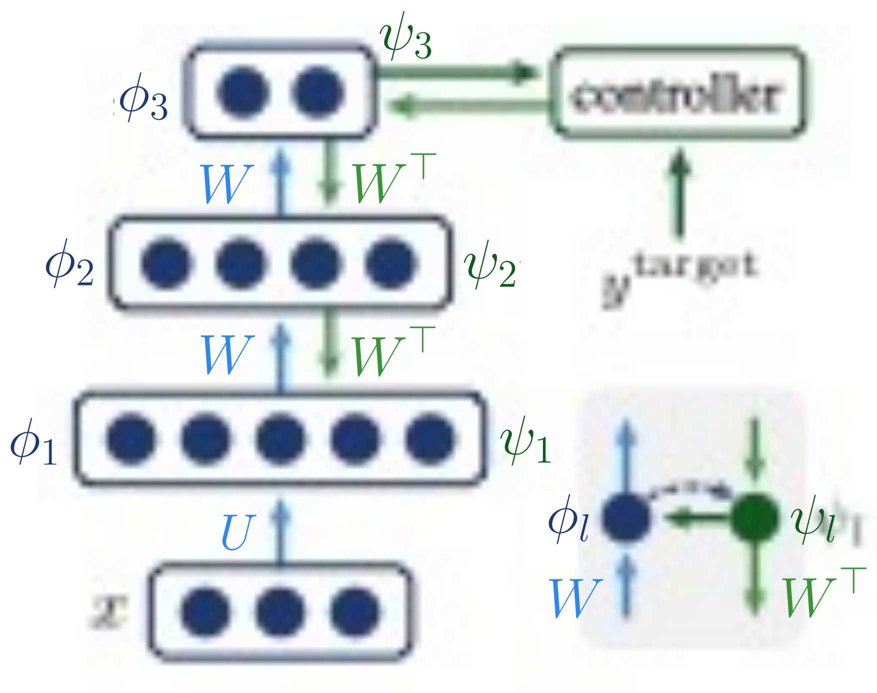
the control signal is directly fed back to hidden states
$\psi = Qu$

Q is learned using local Hebbian rules

approximate optimal control in general, exact for one data point or for linear networks

Lillicrap et al., *Nat. Comm.* 2016; Nokland, *NeurIPS* 2016; Meulemans et al., *NeurIPS* 2021, *ICML* 2022; Akrout et al., *NeurIPS* 2019

Dynamic inversion



network and controller dynamics jointly converge to an exact optimal control

$\dot\phi = -\phi + W\rho(\phi) + Ux + \psi$
$\dot\psi = -\psi + \rho'(\phi)W^\top\psi + u$

"enhanced" version of (R)BP

feedback weights can be learned too

Results

	Feedforward (test acc., %)		**Recurrent** (test acc., %)	
	MNIST	CIFAR-10	MNIST	CIFAR-10
LCP (linear feedback)	97.73 ± 0.07	/	97.70 ± 0.11	/
LCP (dynamic inversion)	98.11 ± 0.07	77.28 ± 0.10	97.58 ± 0.16	80.26 ± 0.17
LCP (dynamic inversion + learned feedback)	98.14 ± 0.09	77.16 ± 0.10	97.75 ± 0.11	/
(R)BP	98.29 ± 0.14	77.58 ± 0.14	97.87 ± 0.19	80.14 ± 0.20
	2-hidden-layer feedforward network	convolutional network	fully-connected equilibrium RNN	convolutional equilibrium RNN