

Self-Repair as a Bounded Disturbance: Input-to-State Stability of Neural Network Training Dynamics

Rokib Al Dhin Raadh Independent researcher · high-school senior, Dhaka, Bangladesh

DeepMath 2026

The Ohio State University
29–30 October 2026

Why this matters now

Networks that train for long lose plasticity: units die and stop learning (Dohare et al., *Nature* 2024). The field's fix is to repair them mid-training: Continual Backprop, ReDo (ICML 2023), Shrink & Perturb (NeurIPS 2020), plasticity injection (NeurIPS 2023). A repair is *not the gradient of any loss*, so the usual convergence proofs cover none of them.

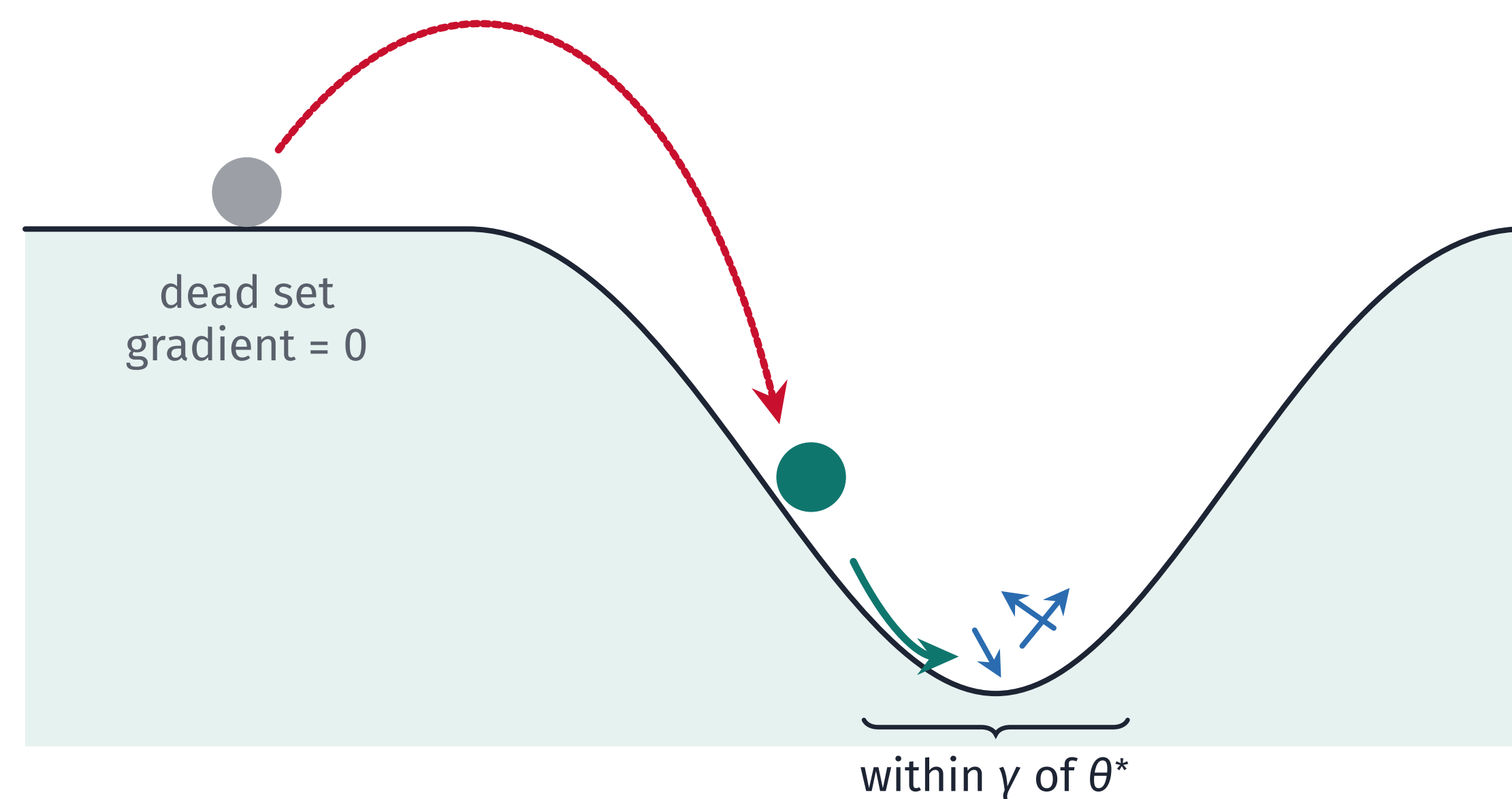
When does a network that fixes itself still converge?

What's new: self-repair modelled as an input to the training dynamics, so ISS bounds for perturbed descent (Cui, Jiang, Sontag & Braatz, 2025) apply to it; and a matching impossibility result that explains why resets are needed.

Repair as a control input

$$\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t) + u_t$$

Treat each repair u_t as an *input* and ask for input-to-state stability (Sontag, 1989). Training becomes a switched system:



flow: the gradient step, contracting near θ^*

flicks: small capped corrections, $\|u_t\| \leq \varepsilon_t \leq \varepsilon_{\max}$

jumps: resets that redraw a dead layer

The repair rule we analyse

Each layer gets a health score from three signals: the global loss, its own local loss through a shared readout, and its gradient stability. An alarm fires when health falls 2.5σ below its own running mean. Corrections are clipped to at most $0.9\|w\|$ (and mean-centred on the DAG and ResNet); a layer dead for three epochs is redrawn.

383 runs: 50-node DAG, two CNNs, ResNet-18; CIFAR-10/100.

NETWORKS ARE NOW REPAIRED MID-TRAINING. WHEN IS THAT SAFE?

Small repairs keep training stable. Only a jump can revive a dead layer.

We prove both, which makes self-repair a switched system: bounded flicks between isolated jumps.

PROPOSITION 1 · PRACTICAL ISS

Bounded repair is safe

Near a minimum θ^* let L be ℓ -smooth and α -strongly convex, $\eta \leq 1/\ell$, repairs $\|u_t\| \leq \varepsilon_t$, local gradients (weight β) bounded by M . While θ_t stays in that neighbourhood,

$$\limsup_{t \rightarrow \infty} \|\theta_t - \theta^*\| \leq \gamma = \frac{2\beta M}{\alpha} + \frac{\bar{\varepsilon}}{\eta\alpha}$$

where $\bar{\varepsilon} = \limsup_t \varepsilon_t$.

Repairs push training around, but never further than γ from a good solution.

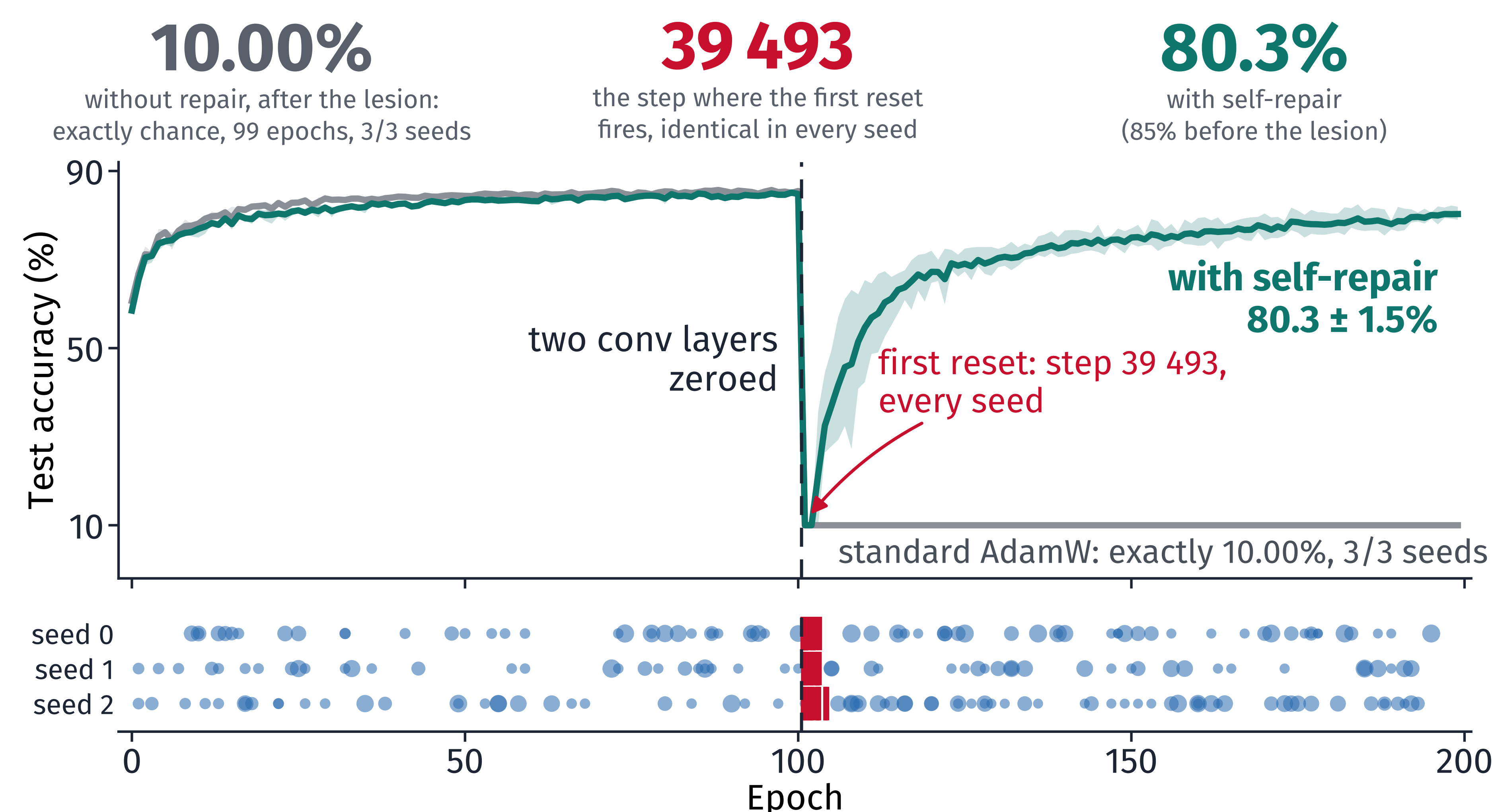
THEOREM 2 · ABSORBING DEAD SET

Bounded repair cannot revive

In a Conv-BN-ReLU layer (BatchNorm in train mode) let channel c have zero filter W_c and BN shift $\beta_c \leq 0$. This dead set D is invariant under AdamW (fresh state) and under every repair bounded per channel, $\|R(W)_c - W_c\| \leq f\|W_c\|$:

$$\theta_0 \in D \implies \theta_t \in D \quad \forall t$$

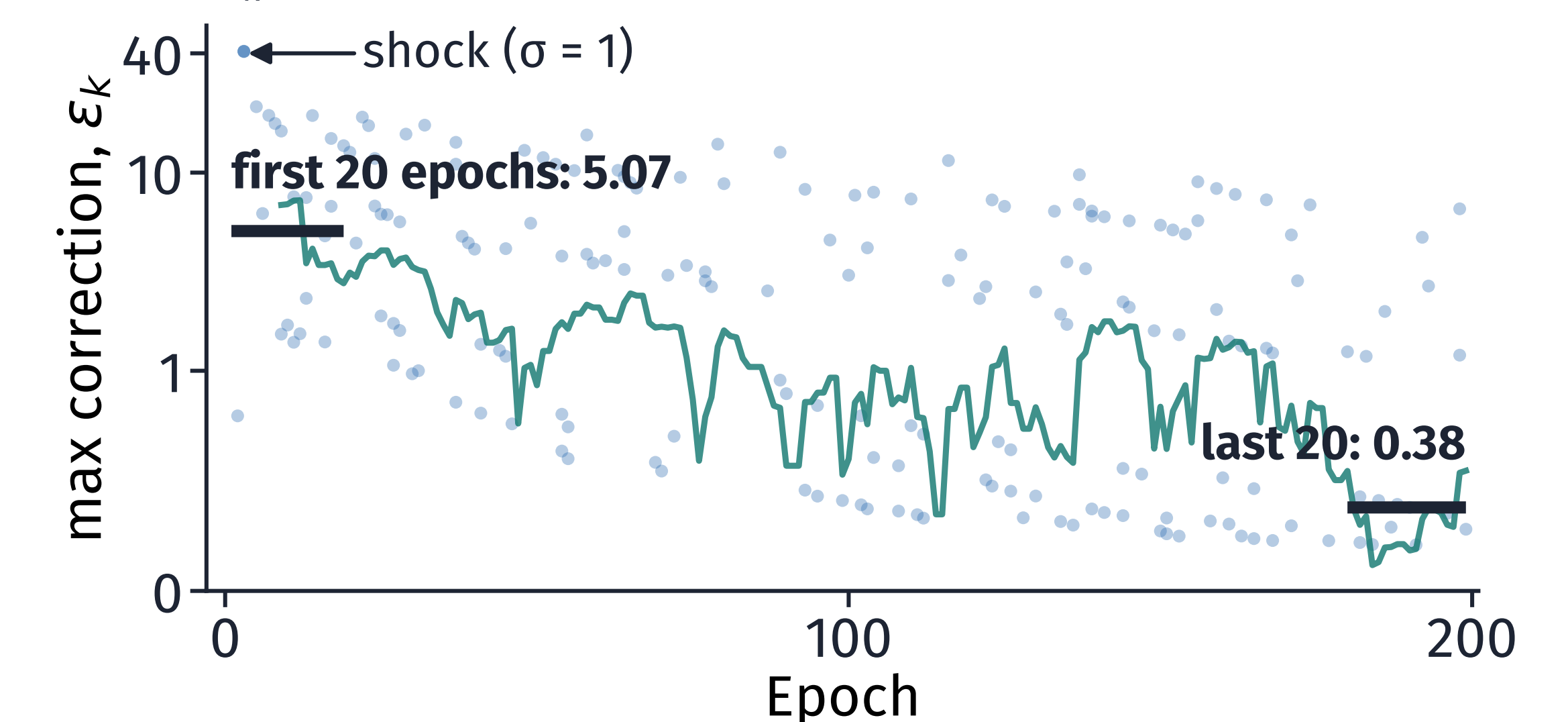
Gradient descent and small fixes are both stuck, exactly. Only a jump leaves D .



Theorem 2, in data. CIFAR-10 CNN; two conv layers zeroed after epoch 100. Small corrections cannot revive them, so accuracy sits at chance until a **reset** (a jump) redraws the dead channels at step 39 493 in every seed; **small corrections** then finish the recovery. Bottom: every repair event.

A falsifiable prediction

Proposition 1's bound depends on $\bar{\varepsilon}$, so it is informative only if repairs *shrink*. They do: after a shock, the largest correction per epoch, ε_k , falls **5.07** $\rightarrow$ **0.38** (**-93%**).



Where repair stops mattering

On ResNet-18 the skip path routes around the same lesion, so the network barely notices it: **92.3%** without repair vs **91.6%** with (3-seed means). Repair matters when there is no redundancy to fall back on.

Proof ideas

Prop. 1. Near θ^* the gradient step contracts by $\rho = 1 - \eta\alpha$, so $r_{t+1} \leq \rho r_t + 2\eta\beta M + \varepsilon_t$; take lim sup: $\limsup r \leq (2\eta\beta M + \bar{\varepsilon})/(\eta\alpha)$.

Thm 2. A zero filter feeds BatchNorm a constant, so $\hat{x} \equiv 0$. With $\beta_c \leq 0$ the ReLU is closed and every gradient into the channel is exactly 0. AdamW then gives $\beta_c(t) = (1 - \eta\lambda)^t \beta_c(0)$, which we see to the bit in the logs.

OPEN PROBLEM

How often may a network jump?

Proposition 1 holds between jumps; Theorem 2 says jumps are unavoidable. Find a dwell time τ_D such that

$$\text{jumps} \geq \tau_D \text{ apart} \implies \limsup_t r_t \leq \gamma'$$

(cf. Hespanha & Morse, 1999). Also open: strong convexity cannot hold exactly for BatchNorm layers (scale invariance); Cui et al. (2025) give ISS under a generalised PL condition, but whether BN networks satisfy it is open; and $\varepsilon(t)$ derived rather than measured.



raadh.me/deepmath

Scan: paper and ask the author

Theorem 2 as a free tool:
pip install ormas-monitor

The author could not travel.
Open to research collaborations;
questions welcome:

raadh@oxiedo.com

ORCID 0009-0003-1178-5296