Gradient Stabilisation

ar015 · 12 June 2026

Conductance-based spiking networks (COBA, PING) need one extra ingredient to train at all: a single flag, `--v-grad-dampen`. This article is the deep dive on why it is needed and what it does, derived from the discrete equations the code runs. For the surrounding recipe — BPTT, the spike surrogate, the loss, optimiser, readout, and regulariser — see the Training article; this one assumes that background.

In plain English. The E and I cells form a loop that makes the network oscillate at gamma (≈25 Hz). Training sends the gradient backward around that same loop, and each time a neuron crosses its spike threshold the surrogate hands the gradient a large multiplier. The loop is traversed *once per gamma cycle* — only about five times in a 200 ms trial — but each pass multiplies by a factor well above one (the surrogate slope enters squared), so a handful of passes is enough to overflow to NaN. The forward simulation stays bounded because the spike reset clamps each cell every cycle; the gradient sidesteps that reset, so forward stability buys nothing backward. The fix, `--v-grad-dampen`, divides the voltage gradient by a constant γ at every step, shrinking the loop's multiplier below one — and thanks to a straight-through trick it changes only the gradient, never the simulation. The rest of this article proves each of those claims from the discrete equations.

Naive BPTT through a 2000-step COBA/PING trial reaches NaN within a few batches. This article establishes the failure and its remedy from the discrete equations the code runs — no step omitted. The plan is: write the one-step map, differentiate it exactly (Lemma 1), propagate the gradient backward (the recursion), show the recurrent loop forces that gradient to diverge geometrically (Proposition 1), and show that scaling the per-step voltage gradient by $1/\gamma$ makes the loop a contraction while leaving the forward trajectory untouched (Proposition 2).

Setup: the discrete update

Index the timesteps $t = 0, ..., T$ at $\Delta t = 0.1$ ms. Each cell holds a state (V^t, g_e^t, g_i^t). With reversal potentials E_L, E_e, E_i, capacitance C_m, leak g_L, synaptic decays $\beta_e = e^{-\Delta t/\tau_e}$ and $\beta_i = e^{-\Delta t/\tau_i}$, threshold ϑ and reset V_r, the step is exactly (`lif_step_expeuler`, `exp_synapse`):

$$s^t = \Theta(V^t - \vartheta)\chi^t, \quad \text{(S)}$$

$$g_e^{t+1} = \beta_e\big(g_e^t + W_e s_{\text{pre}}^t\big), \quad g_i^{t+1} = \beta_i\big(g_i^t + W_i s_{\text{pre}}^t\big), \quad \text{(C)}$$

$$\tilde{V}^{t+1} = V_\infty^{t+1} + (V^t - V_\infty^{t+1})\alpha^{t+1}, \quad V^{t+1} = \begin{cases} V_r & \text{if } \ s^t = 1 \ \text{ or refractory} \\ \tilde{V}^{t+1} & \text{otherwise,} \end{cases} \quad \text{(V)}$$

where $\chi^t \in \{0, 1\}$ is the not-refractory gate, and the membrane decay and rest point are built from the *post*-update conductances g^{t+1}:

$$g_{\text{tot}} = g_L + g_e^{t+1} + g_i^{t+1}, \quad \alpha^{t+1} = e^{-\Delta t g_{\text{tot}}/C_m}, \quad V_\infty^{t+1} = \frac{g_L E_L + g_e^{t+1} E_e + g_i^{t+1} E_i}{g_{\text{tot}}}. \quad \text{(P)}$$

The timing matters and is taken from the code: the spike s^t in (S) is read from the *current* voltage V^t, drives the conductance one step later in (C), and that conductance sets the voltage in (V). So one synaptic edge — presynaptic voltage to postsynaptic voltage — spans one timestep.

Lemma 1 — the one-step Jacobian

Differentiate (S), (C), (V) entry by entry. The spike surrogate gives the spike derivative

$$\frac{\partial s^t}{\partial V^t} = \sigma'(V^t - \vartheta)\chi^t, \quad \sigma'(u) = \frac{k}{(1 + k\,|u|)^2}, \quad \sigma'(0) = k. \quad (1)$$

From (C), the conductance partials are a self-decay and a *presynaptic-voltage* coupling obtained by chaining (1):

$$\frac{\partial g_e^{t+1}}{\partial g_e^t} = \beta_e, \quad \frac{\partial g_e^{t+1}}{\partial V_{\text{pre}}^t} = \beta_e W_e \sigma'\big(V_{\text{pre}}^t - \vartheta\big)\chi_{\text{pre}}^t. \quad (2)$$

From (V), with g held, the membrane self-term is the decay; and differentiating (P) gives the conductance-to-voltage term ($\partial g_{\text{tot}}/\partial g_e = 1$, $\partial V_\infty/\partial g_e = (E_e - V_\infty)/g_{\text{tot}}$, $\partial \alpha/\partial g_e = -(\Delta t/C_m)\alpha$):

$$\frac{\partial \tilde{V}^{t+1}}{\partial V^t} = \alpha^{t+1}, \quad \frac{\partial \tilde{V}^{t+1}}{\partial g_e^{t+1}} = \underbrace{(1-\alpha)\frac{E_e - V_\infty}{g_{\text{tot}}}}_{\text{rest shifts}} - \underbrace{\frac{\Delta t}{C_m}\alpha(V^t - V_\infty)}_{\text{decay shifts}} \equiv \kappa_e \approx \frac{\Delta t}{C_m}(E_e - V), \quad (3)$$

the last step being the leading order in Δt (using $1 - \alpha \approx \Delta t g_{\text{tot}}/C_m$). Define κ_i identically with E_i. Collecting (1)–(3), the one-step Jacobian on (V, g_e, g_i), with the cross-cell coupling in the lower-left block, is

$$J^t = \frac{\partial (V, g_e, g_i)^{t+1}}{\partial (V, g_e, g_i)^t} = \begin{pmatrix} \rho\alpha & \kappa_e & \kappa_i \\ \beta_e W_e \sigma'\big(V_{\text{pre}} - \vartheta\big)\chi_{\text{pre}} & \beta_e & 0 \\ \beta_i W_i \sigma'\big(V_{\text{pre}} - \vartheta\big)\chi_{\text{pre}} & 0 & \beta_i \end{pmatrix}. \quad \text{(J)}$$

The single subtlety, and it is decisive below: the reset in (V) is a `torch.where`, so on any cell that spikes or is refractory the output is the constant V_r and the membrane self-term is gated to zero. We write this as the factor $\rho \in \{0, 1\}$ on the $\partial V^{t+1}/\partial V^t$ entry: $\rho = 0$ at a spike (gradient through the cell's own membrane is cut), $\rho = 1$ otherwise. Crucially ρ multiplies only the membrane self-term — it does **not** touch the spike output s^t in (1)–(2), which is evaluated at V^t *before* the reset.

Backpropagation: the gradient recursion

Let $\lambda^t \equiv \partial \mathcal{L}/\partial (V, g_e, g_i)^t$. Reverse-mode autodiff is exactly the linear recursion

$$\lambda^t = \big(J^t\big)^\top \lambda^{t+1}, \quad \lambda^T = \nabla_{x^T}\mathcal{L} \quad \Longrightarrow \quad \lambda^t = \left(\prod_{s=t}^{T-1} J^s\right)^\top \lambda^T. \quad (4)$$

So the gradient that reaches step t is governed by the product of one-step Jacobians, and $\|\lambda^t\| \le \left(\prod_s \|J^s\|\right) \|\lambda^T\|$. Whether this is benign or catastrophic is decided by the voltage component of (J) chained through the recurrent wiring.

Proposition 1 — the backpropagated gradient diverges (the problem)

Claim. In a network with a recurrent E→I→E loop, the voltage gradient grows geometrically in the number of gamma *cycles* traversed — not in the number of timesteps. A 200 ms trial holds only $N \approx 5$ cycles against $T = 2000$ steps; the danger is that each cycle multiplies by a large factor, not that there are many cycles.

Proof. Compose (2) and (3): the gradient carried from a postsynaptic voltage at $t\{+\}1$ to a presynaptic voltage at t across one synapse is the product of the conductance-coupling and the membrane term,

$$a \equiv \frac{\partial V_{\text{post}}^{t+1}}{\partial V_{\text{pre}}^{t}} = \underbrace{\kappa}_{\approx \frac{\Delta t}{C_m}(E_{\text{syn}} - V)} \cdot \underbrace{\beta W}_{\text{synapse}} \cdot \underbrace{\sigma'\big(V_{\text{pre}} - \vartheta\big)}_{\text{spike}}. \quad (5)$$

Traverse the loop once: $V_E \to V_I$ across W_{ei} (edge gain a_{ei}), then $V_I \to V_E$ across W_{ie} (edge gain a_{ie}). Over one round trip the E-voltage gradient maps to itself with the **loop gain**

$$\varrho = a_{ei} a_{ie} = \underbrace{k^2}_{\text{two } \sigma' \text{ at volley}} \cdot \beta_e \beta_i W_{ei} W_{ie} \kappa_I \kappa_E. \quad (6)$$

The factor k^2 appears **once per gamma cycle**. Each population fires a single synchronous volley per cycle, so the two large kicks $\sigma' \to \sigma'(0) = k$ (one for E, one for I) occur together once per loop traversal and nowhere else; between volleys the per-step Jacobians are the mild sub-unit decays $\alpha, \beta < 1$, which merely carry the gradient along without amplifying it. So (4) collapses, across cycles, to the scalar recursion $\lambda_{V,E}^{(n)} \approx \varrho \lambda_{V,E}^{(n+1)}$, giving after N cycles

$$|\lambda_{V,E}| \geqslant |\varrho|^N \, |\lambda_{V,E}^T|. \quad (7)$$

Worked example (default PING init). Take the released constants: $\Delta t = 0.1$ ms, $k = 5$, $C_m^E = 1.0$ nF, $C_m^I = 0.5$ nF, $E_e = 0$, $E_i = -80$ mV, $V \approx \vartheta = -50$ mV at the crossing, $\tau_{\text{AMPA}} = 2$ ms ($\beta_e = e^{-0.05} \approx 0.95$), $\tau_{\text{GABA}} = 9$ ms ($\beta_i \approx 0.99$), and order-µS coupling $W_{ei} \approx 1$, $W_{ie} \approx 2$ µS. The two edge gains (5) are

$$a_{ei} = \underbrace{\frac{0.1}{0.5}(0 - (-50))}_{\kappa_{\to I} = 10} \cdot \underbrace{0.95 \cdot 1}_{\beta_e W_{ei}} \cdot \underbrace{5}_{k} \approx 48,$$

$$a_{ie} = \underbrace{\frac{0.1}{1.0}(-80 - (-50))}_{\kappa_{\to E} = -3} \cdot \underbrace{0.99 \cdot 2}_{\beta_i W_{ie}} \cdot \underbrace{5}_{k} \approx -30,$$

so the loop gain is $\varrho = a_{ei} a_{ie} \approx -1.4 \times 10^3$. Over the $N \approx 5$ cycles of a 200 ms trial — not the $T = 2000$ steps — the voltage gradient is amplified by

$$|\varrho|^N \approx \left(1.4 \times 10^3\right)^5 \approx 6 \times 10^{15}.$$

A unit gradient seeded at the readout thus returns to $t = 0$ scaled by $\sim 10^{16}$; summed over the batch and compounded across successive optimiser steps it crosses the fp32 ceiling (≈

3.4×10^{38}) and the loss becomes NaN within a few batches. The blow-up is robust: even if threshold spread cuts the effective σ' tenfold (each edge $\div 10$, so $\varrho \div 100$), $|\varrho| \approx 14$ and $|\varrho|^5 \approx 6 \times 10^5$ — still divergent. The compounding is per cycle, not per step. ■

Why the forward pass does not blow up the same way. The forward orbit is bounded because the reset (V) slams each spiking cell to V_r every cycle — that is the $\rho = 0$ gate in (J), a strong per-cycle contraction. But ϱ in (6) is built only from the spike-output edges (5), i.e. from σ' evaluated *before* the reset; the gate ρ sits on the membrane self-term and never enters the loop product. So the backward loop bypasses precisely the contraction that bounds the forward orbit. Forward stability and backward divergence are not in contradiction: they travel different paths through (J), and the straight-through reset is what separates them.

Proposition 2 — per-step voltage-gradient damping bounds it (the solution)

The flag `--v-grad-dampen` inserts, before the reset, the operation `dv = _scale_grad(dv, 1/γ)` where $dv = (V_\infty - V)(1-\alpha)$ is the membrane increment in (V). The primitive

$$\text{\_scale\_grad}(x, c) = cx + \text{detach}(x)(1-c) \quad (8)$$

is the identity in the forward pass ($cx + (1\{-\}c)x = x$) but multiplies the backward gradient through x by $c = 1/\gamma$. Re-differentiating (V) with dv so scaled changes two partials of (J):

$$\frac{\partial V^{t+1}}{\partial V^t} = 1 - \frac{1-\alpha}{\gamma} \in [\alpha, 1], \quad \frac{\partial V^{t+1}}{\partial g^{t+1}} = \frac{\kappa}{\gamma}. \quad (9)$$

The membrane self-term stays bounded by 1 (for $\gamma \to \infty$ it tends to 1 — the slow integration pathway is *preserved*, not crushed), while every voltage←conductance edge — and therefore every loop edge (5) — is divided by γ. The loop gain (6) becomes

$$\varrho_\gamma = \frac{a_{ei}}{\gamma}\frac{a_{ie}}{\gamma} = \frac{\varrho}{\gamma^2}, \quad (10)$$

so the backward loop is a contraction, $|\varrho_\gamma| \le 1$, as soon as

$$\gamma \ge \sqrt{|\varrho|}. \quad (11)$$

By (8) the forward trajectory x^t is bitwise identical with or without the flag, so this is a pure modification of the gradient, not of the dynamics. ■

In code this is the single line `_scale_grad(dv, 1.0 / v_grad_dampen)` in the LIF step. Since $|\varrho| \sim k^2 c$ for an order-unity loop constant c, the threshold (11) scales like $\sqrt{|\varrho|} \sim k\sqrt{c}$, consistent with the recipes: $\gamma \approx 80$ for the unitless standard SNN and $\gamma \approx 1000$ for COBA/PING (used by exp025 and downstream), whose larger conductance-scale loop constant demands the larger γ.

Corollary — the cost, and choosing γ

The $1/\gamma$ in (9) lands on *every* voltage←conductance gradient, not only the recurrent loop. The feedforward input also enters through g_e (the term $W_{\text{in}} s_{\text{in}}$ in (C)), so the input-weight gradient is suppressed by the same factor: damping trades a slice of the legitimate learning signal for stability. Hence the operating rule — take the smallest γ satisfying (11), i.e. the smallest value that prevents overflow. Too large a γ can therefore quietly cap achievable

accuracy by starving the input layer of gradient. Distinct from gradient clipping, which rescales the assembled parameter gradient after the fact: damping reshapes the recursion (4) term by term, before any parameter gradient is formed. Tightening (11) per-layer on long-trial tasks is open work.

Prediction. The mechanism implies the flag is load-bearing only when the loop exists: the same network should train with damping fully off when run as COBA ($W_{ei} = 0$, loop open), but fail as PING ($W_{ei} > 0$) — every optimiser step's gradient going non-finite and being skipped, leaving the network frozen at chance.