# Critical Neural Networks in Atari Games

Thomas Pluck[1] and Aaron McAfee[1]
[1]Department of Electronic Engineering
Maynooth University
Maynooth, Ireland
Email: {thomas.pluck.2025, aaron.mcafee.2021}@mumail.ie

***Abstract*—**

## I. Introduction

## II. Background

### A. Reinforcement Learning

### B. Statistical Mechanics

Criticality in statistical mechanics refers to phase transitions between states of matter. At criticality, systems exhibit "scale-free" correlations where microscopic and macroscopic scales become indistinguishable, manifesting as power-law distributions and long-range correlations that span all space and time scales of the system.

The Ising model is often used in studying criticality, where atoms in a lattice have spin states $s_i \in \{+1, -1\}$. Each atom $s_i$ couples to others with parameter $J_{ij}$ and experiences local bias from a field $h_i$. The energy of a configuration $\vec{s}$ is:

$$E(\vec{s}) = \sum_{i \neq j} J_{ij} s_i s_j + \sum_i h_i s_i \tag{1}$$

This defines a Gibbs distribution over the states $\vec{s}$:

$$\mathbb{P}(\vec{s}) = \frac{1}{Z} \exp(-\beta E(\vec{s})) \tag{2}$$

with partition function $Z$ and inverse pseudotemperature $\beta$.

For neural systems, the mean-field approximation of highly connected Ising models is particularly relevant. In this approximation, the expected value of each spin is:

$$\langle s_i \rangle = \tanh(\beta(\sum_j J_{ij} \langle s_j \rangle + h_i)) \tag{3}$$

Which can be noted to bear a striking similarity to a hyperbolic tangent artificial neuron:

$$\sigma_i = \tanh\left(\sum_j w_{ij} x_j + b_i\right) \tag{4}$$

Renormalization Group (RG) analysis allows us to understand scale-invariant behavior by coarse-graining (ie. averaging) the individual spins into blocks of size $b^d$ with block-spin variables given by:

$$S_I = \frac{1}{b^d} \sum_{i \in \text{block } I} s_i \tag{5}$$

Under this transformation, the effective parameters of the system (coupling strengths $J_{ij}$ and fields $h_i$) change according to RG flow equations $dS_I/db$. A system exhibits critical behavior when its parameters are tuned to values that remain invariant under this transformation - specifically at the non-trivial fixed point of these flow equations (ie. $dS_I/db = 0$) which form a dense subset in parameter space and construct a critical manifold.

### C. Criticality in Neural Systems

The criticality hypothesis posits that biological neural systems self-organize to operate near critical points between ordered and chaotic dynamics [1], [2]. Empirical evidence includes observations of "neuronal avalanches" in cortical tissue with size distributions following power laws with exponents of approximately -3/2, matching predictions from critical branching processes [1].

Neural networks near criticality demonstrate optimal computational properties, including maximized dynamic range [3], [4], information transmission [2], and information storage capacity [5]. Conversely, deviations from criticality correlate with neural pathologies [6], suggesting that maintaining criticality is essential for healthy brain function.

These findings motivate our approach: rather than training networks that may accidentally drift away from criticality, we leverage RG flow analysis to design networks that intrinsically maintain critical dynamics throughout operation.

## III. Methodology

### A. Proposed Implicit Regularizer

Our proposal primarily builds on the Anti-Correlated Noise Injection (ACNI) implicit regularization method [7]. In ACNI, standard stochastic gradient descent an i.i.d noise source $\xi_n \sim \mathcal{N}(\vec{0}, \sigma^2 I)$ is used to create temporally anticorrelated noise $\zeta_n = \xi_n - \xi_{n-1}$ - this is then fed into the gradient step on parameter $w_n$:

$$w_{n+1} = w_n - \eta \nabla L(w_n) + \zeta_n \tag{6}$$

It is subsequently proven in the ACNI paper that an implicit loss function emerges from the action perturbations on gradients that minimizes the trace Hessian leading to robust, flat region of the loss landscape:

$$\tilde{L}(z) = L(z) + \frac{\sigma^2}{2} \text{Tr}(\nabla^2 L(z)) \tag{7}$$

Our main contribution is the introduction a control mechanism to the variance term $\sigma^2$ in ACNI that works to bias training toward edge-the-chaos and criticality.

The variance that we define is given by:

$$\sigma^2 = \frac{2}{\sqrt{N}} \left( \frac{1}{N} - \frac{1}{\|\nabla L(z)\|} \right) \tag{8}$$

Explicitly this leads to the implicit loss function:

$$\tilde{L}(z) = L(z) + \frac{1}{\sqrt{N}} \Bigg( \underbrace{\frac{1}{N}}_{\substack{\text{quadratic}\\\text{scaling}}} - \underbrace{\frac{1}{\|\nabla L(z)\|}}_{\substack{\text{linear}\\\text{scaling}}} \Bigg) \mathrm{Tr}(\nabla^2 L(z)) \tag{9}$$

Where $N$ is the rank of the learned map. This quantity is explicitly derived from a minimization problem with a contrived solution at the edge of chaos. Surprisingly, this includes two competing terms that scale quadratically and linearly respectively - implying that the minimization of this term can only come about in regimes where quadratic and linear scaling are equal, ie. in a scale-free region of the loss-landscape. Complete derivation of this term and proof that scale-free loss implies that ANN models learn scale-free maps can be found in the appendix.

### B. Enviroment Setup

### C. Experiments

## IV. Results

## V. Discussion

## VI. Conclusion

## Acknowledgment

The authors would like to thank the instructors and teaching assistants of CS637 for their guidance and support throughout this project. We extend our gratitude to our peers for their valuable feedback during presentation rehearsals and draft reviews. Special thanks to Barak Pearlmutter for introducing us to key concepts in reinforcement learning that formed the foundation of this work. We also acknowledge the resources provided by our university's computing facilities that enabled our experimental implementations. Finally, we thank our families and friends for their unwavering support throughout our academic journey.

## References

[1] J. M. Beggs and D. Plenz, "Neuronal avalanches in neocortical circuits," *Journal of Neuroscience*, 2003.

[2] J. M. Beggs and N. Timme, "Being critical of criticality in the brain," *Front. Physio.*, 2012.

[3] O. Kinouchi and M. Copelli, "Optimal dynamical range of excitable networks at criticality," *Physical Review E*, 2006.

[4] W. L. Shew, H. Yang, T. Petermann, R. Roy, and D. Plenz, "Neuronal avalanches imply maximum dynamic range in cortical networks at criticality," *Journal of Neuroscience*, 2009.

[5] N. Bertschinger and T. Natschläger, "Real-time computation at the edge of chaos in recurrent neural networks," *Neural Computation*, 2004.

[6] C. Meisel, A. Storch, S. Hallmeyer-Elgner, E. Bullmore, and T. Gross, "Failure of adaptive self-organized criticality during epileptic seizure attacks," *PLoS Comput. Biol.*, 2011.

[7] A. Orvieto, H. Kersting, F. Proske, F. Bach, and A. Lucchi, "Anti-correlated noise injection for improved generalization," in *International Conference on Machine Learning*. PMLR, 2022, pp. 17 094–17 116.

## Appendix

In this appendix, we provide a rigorous derivation of our proposed regularization term that promotes criticality in neural networks. Our approach drives networks to the edge of chaos through both explicit Jacobian constraints and implicit scale-free dynamics.

### A. Regularizing to the Edge of Chaos

Let us define a standard feedforward network $a = \sigma(z)$ where preactivation $z = Wx + b$ is defined with weight matrix $W$, bias $b$, input $x$ and the put through non-linearity $\sigma$ to create activation $a$. A known fact about rank-$N$ operators $J$ is that their Lyapunov exponent collapse when $\|J\|_F^2 = N$ at the so-called "edge of chaos".

We derive our regularizer by letting $J$ be the Jacobian of the feedforward layer $a = \sigma(z)$ and finding explicit derivatives in terms of weights $W$ and biases $b$ to minimize the quantity $J$ - to simplify derivation we will focus entirely on individual entries of $b$ the $b_i$ and assure the reader that much same terms will arise when computing the derivative of $W_{ij}$

Let us begin by computing the derivative of $a_i$ with respect to $x_j$ for the individual terms of the Jacobian $J_{ij}$:

$$J_{ij} = \frac{\partial}{\partial x_j} \sigma(z_i) = W_{ij} \sigma'(z_i) \tag{10}$$

We can now compute the Frobenius norm of $J$ and begin computing it's derivative w.r.t. $b_i$:

$$\frac{\partial}{\partial b_i} \|J\|_F = \frac{\partial}{\partial b_i} \sqrt{\sum_{i,j} W_{ij}^2 \sigma'(z_i)^2} \tag{11}$$

$$= \frac{\sum_j W_{ij}^2 \sigma'(z_i) \sigma''(z_i)}{\|J\|_F} \tag{12}$$

We note at this juncture that $\frac{\partial^2}{\partial x_j^2} \sigma(z_i) = W_{ij}^2 \sigma''(z_i)$ so we may write:

$$\frac{\partial}{\partial b_i} \|J\|_F = \frac{\sigma'(z_i) \nabla^2 \sigma(z_i)}{\|J\|_F} \tag{13}$$

Where $\nabla^2$ is the Laplace operator. We would now like to encode the edge of chaos criterion $\|J\|_F^2 = N$ into an explicit quantity that we can minimize using the parameters of our network.

$$\frac{\partial}{\partial b_i} \left( 1 - \frac{\|J\|_F}{\sqrt{N}} \right)^2 = \frac{\partial}{\partial b_i} \left( 1 - \frac{2\|J\|_F}{\sqrt{N}} + \frac{\|J\|_F^2}{N} \right) \tag{14}$$

$$= 2 \frac{\partial}{\partial b_i} \|J\|_F \cdot \frac{\|J\|_F}{N} - \frac{\partial}{\partial b_i} \frac{2\|J\|_F}{\sqrt{N}} \tag{15}$$

$$= \frac{2\sigma'(z_i) \nabla^2 \sigma(z_i)}{\sqrt{N}} \left( \frac{1}{N} - \frac{1}{\sqrt{N} \|J\|_F} \right) \tag{16}$$

With the derivation complete, we can note a few things about this quantity...