Continuous Sparsification via Minimizing Movement

Hoang Pham HOANG.PHAM@WARWICK.AC.UK
University of Warwick

Tom Jacobs TOM.JACOBS@CISPA.DE
CISPA Helmholtz Center for Information Security

Binh Nguyen BINH.NT2@VINUNI.EDU.VN
VinUniversity

Rebekka Burkholz BURKHOLZ@CISPA.DE
CISPA Helmholtz Center for Information Security

Long Tran-Thanh LONG.TRAN-THANH@WARWICK.AC.UK
University of Warwick

Abstract

Continuous sparsification is a popular approach for finding efficient sparse subnetworks, optimizing soft masks before applying a hard Top-K projection. However, existing methods typically regularize mask variables using flat Euclidean geometry, penalizing local rewirings and disruptive long-range relocations equally. To address this, we propose Continuous Sparsification via Minimizing Movement (CSMM), a geometry-aware framework that treats layer connectivity as a probability allocation on the simplex. By leveraging the minimizing-movement scheme, we regularize the temporal evolution of connectivity using flexible proximal penalty functions. This approach decouples task performance from structural evolution, allowing practitioners to impose specific geometric inductive biases on topology evolution. The experiment shows a promising result on CIFAR10 dataset under ResNet-20 architecture, offering a new approach on continuous sparsification.

1. Introduction

Modern deep networks are often over-parameterized, making sparse training essential for memory and compute efficiency [10, 13, 20]. While Lottery Ticket methods [5, 6, 10] achieve high performance, they incur significant iterative training costs. Alternatively, Dynamic Sparse Training (DST) [9, 23, 28] evolves topology during training from scratch. Interestingly, Sparse Evolutionary Training [28] empirically shows that network connectivity naturally evolves from random, uniform initializations into scale-free topologies. Viewed through the lens of graphons [2, 26, 31, 32], the continuous limit representations of large networks, this phenomenon describes a fundamental topological transition: the evolution from a flat, homogeneous graphon to a inhomogeneous one.

Despite this elegant geometric reality, existing methods fail to model this transition smoothly. DST approaches drive this evolution through abrupt, discrete prune-and-regrow heuristics, rather than a principled optimization objective. Conversely, continuous sparsification methods [16, 19, 33, 39] avoid discrete updates by optimizing soft mask variables, but they treat these masks as unconstrained, flat vectors. By relying on standard Euclidean updates, they ignore the structural geometry of the neural architecture, penalizing minor local rewirings and long-range allocations equally.

© H. Pham, T. Jacobs, B. Nguyen, R. Burkholz & L. Tran-Thanh.

Motivated by the continuous sparsification and graphon perspective, we reframe sparse training as a process of continuous mass evolution. We replace discrete binary masks with a finite-dimensional connectivity allocation ρ: a probability distribution over potential edges. By constraining ρ to the probability simplex, we natively transform discrete rewiring into continuous, constrained optimization. Rather than abruptly pruning, this allows the network to smoothly transport "connectivity mass" from a homogeneous state into a structured, high-performing topology.

To rigorously control the dynamics of this evolution, we propose Continuous Sparsification via Minimizing Movement (CSMM). We formulate connectivity evolution as a geometric proximal optimization step. At each iteration, we update the weights and the connectivity allocation by minimizing the task energy alongside a distance penalty that penalizes deviations from the previous topology. Crucially, this framework allows for a flexible family of proximal penalties, including L_p norms, KL divergence, and Wasserstein distances, enabling the imposition of new specific inductive biases on how the topology evolves. By constraining the "movement" of connectivity, we decouple task performance from network configuration, turning sparse training into a controlled dynamical system.

Our main contributions are as follows: (1) We introduce a formulation of continuous sparse training that represents layer connectivity as a continuous probability allocation on the simplex. (2) Building on this view, we propose a geometry-aware proximal framework that regularizes connectivity evolution via plug-in proximal penalties, providing a flexible mechanism to control the topological evolution. (3) We analyze the idealized proximal dynamics in a finite-dimensional setting, establishing existence of minimizers together with a one-step descent property and bounded cumulative dissipation, which imply vanishing updates over time. (4) Empirically, we show that our CSMM has promising results against baselines, and different proximal penalties induce distinct accuracy-sparsity trade-offs. These results suggest that the choice of geometry plays a meaningful role in continuous sparsification.

2. Related work

Pruning neural networks . Neural network pruning aims to reduce model size and computational cost by removing parameters while preserving performance [13, 15, 20]. This broad literature includes classical post-training [13, 20] and iterative pruning methods [5, 10, 38], as well as sparse learning approaches [9, 28]. Following the Lottery Ticket Hypothesis [4, 5, 10] , significant effort has focused on identifying trainable sparse subnetworks at initialization [11, 14, 21, 35, 36] and understanding their structural biases through path-aware [12, 29, 30] or infinite-width graphon perspectives [31, 32]. In parallel, sparse training from scratch and dynamic sparse training (DST) methods [8, 9, 24, 28, 37] demonstrated that sparse connectivity can be evolved during training under a fixed parameter budget, often without first training a dense model. These methods typically rely on discrete prune-and-regrow heuristics driven by weight magnitude, momentum, or gradient information. Unlike prior pruning or DST methods, we replace discrete rewiring heuristics with a principled proximal optimization problem over an allocation variable.

Continuous and differentiable sparsification .To avoid discrete mask updates, a complementary line of work utilizes continuous or differentiable surrogates. This includes differentiable L_0 regularization via stochastic gates [25]. Continuous sparsification methods [19, 33] learn continuous mask variables during training and extract hard sparse subnetworks afterwards. Later, [39] proposed spred algorithm, showing that optimizing weight and mask simultaneously, along with weight decay

solve a LASSO objective. PiLoT [16] further extended spred with a mirror flow framework that dynamically adapts the weight decay during training, theoretically illustrates the transition from an implicit L_2 to L_1 regularization. While our framework similarly optimizes a continuous object to extract a hard mask at evaluation, prior methods treat mask parameters as unconstrained variables updated via standard Euclidean steps. In contrast, our approach explicitly regularizes the *temporal evolution* of a simplex-valued connectivity allocation. By applying chosen dissipation functions, we allow practitioners to impose specific geometries on how connectivity mass shifts during training.

Optimization-based sparse training and geometry-aware methods . Our work also builds upon non-Euclidean optimization for sparse networks. Recent works have utilized stochastic Bregman iterations and mirror descent [3, 17, 27] to induce sparsity organically, while others formulate pruning as an Optimal Transport (OT) problem to achieve size-controlled compression [22, 34]. Mathematically, our framework is rooted in the proximal variational formulations of Jordan, Kinderlehrer, and Otto (JKO) [18] and gradient flows in metric spaces [1]. However, rather than using transport primarily as a soft top-k mechanism or relying on standard mirror descent, we introduce a generalized minimizing-movement framework over a simplex-valued allocation. This uniquely allows plug-in dissipations - such as L_1, L_2, KL, and Wasserstein distances - to explicitly govern the evolution of network's connectivity across training.

3. Preliminaries

3.1. Minimizing-movement schemes

Iterative optimization often balances task-objective minimization with stability constraints on the iterates. This is formally captured by the *minimizing-movement* (or proximal) scheme: given an energy E and a state space $\mathcal{X}$, we construct a sequence $\{x^k\}$ via

$$x^{k+1} \in \arg\min_{x\in\mathcal{X}} \; E(x) \; + \; \frac{1}{h}\, D(x, x^k), \tag{1}$$

where $h > 0$ is the step size and $D(\cdot,\cdot)$ is a dissipation function. While D is typically the squared L_2-norm, modern optimization leverages specialized dissipations to encode domain-specific inductive biases—e.g., KL divergence for information geometry or optimal transport for structural locality. This provides a principled mechanism to regularize the *dynamics* of optimization by controlling how much the state is permitted to "move" relative to the chosen geometry.

3.2. Problem setup

We represent the connectivity of a target layer by an allocation ρ belonging to the probability simplex $\Delta \subset \mathbb{R}^{m\times n}$ where m, n are hidden dimensions. To formalize the evolution of ρ, we equip Δ with a metric $d(\cdot,\cdot)$ that defines the cost of moving mass between potential connections. This transforms the connectivity state space into a metric space (Δ, d).

We define the learning dynamics as a joint minimizing-movement scheme over the allocation $\rho \in \Delta^{m\times n}$, and the parameters $\theta \in \mathbb{R}^p$ where p is the number of parameters in a layer (normally $p = m \times n$). Given a step size $h > 0$, a proximal weight $\alpha > 0$, and a dissipation functional $D(\rho, \rho^k) \approx \frac{1}{2} d(\rho, \rho^k)^2$, the next iterate is:

$$(\rho^{k+1}, \theta^{k+1}) \in \arg\min_{\rho\in\Delta, \theta\in\mathbb{R}^p} \; E(\rho,\theta) + \frac{1}{h} D(\rho,\rho^k) + \frac{\alpha}{2h}\|\theta - \theta^k\|_2^2. \tag{2}$$

The choice of the metric space (Δ, d) is the primary tool for controlling structural evolution: (i) *Euclidean geometry* chooses d as the L_2 distance induces a standard Hilbertian geometry, promoting smooth, uniform drift in allocation mass; (ii) *information/statistical geometry* chooses d induced by the KL divergence reflects the information-theoretic distance between allocations, suitable for probabilistic interpretations; and (iii) *optimal transport geometry:* Choosing d as the Wasserstein distance W_p incorporates a ground-cost matrix between connections, allowing us to enforce locality or structural constraints in how "mass" migrates during sparsity evolution.

4. Methodology

4.1. Theoretical analysis

We analyze the idealized proximal update Eq. 2 on the finite-dimensional space $\Delta_{m,n} \times \mathbb{R}^p$. We establish that these dynamics are well-posed and exhibit fundamental descent properties under minimal regularity conditions.

Assumption 1 *E is lower semicontinuous and bounded below on $\Delta_{m,n} \times \mathbb{R}^p$.*

Assumption 2 *For any $\bar{\rho} \in \Delta_{m,n}$, the mapping $\rho \mapsto D(\rho, \bar{\rho})$ is nonnegative, lower semicontinuous, and satisfies $D(\bar{\rho}, \bar{\rho}) = 0$.*

Provided the loss and regularizers are continuous, Asm. 1 is trivially satisfied by our soft gate $S(\rho)$, while Asm. 2 covers L_p norms, KL divergence, and Wasserstein distances.

Theorem 1 *Under Asm. 1 and 2, given any iterate (ρ^k, θ^k) with $E(\rho^k, \theta^k) < \infty$ and $h, \alpha > 0$, the minimization problem (2) admits at least one minimizer $(\rho^{k+1}, \theta^{k+1})$.*

Proposition 2 *Any minimizer $(\rho^{k+1}, \theta^{k+1})$ of (2) satisfies the descent inequality:*

$$E(\rho^{k+1}, \theta^{k+1}) + \frac{1}{h} D(\rho^{k+1}, \rho^k) + \frac{\alpha}{2h} \|\theta^{k+1} - \theta^k\|_2^2 \leq E(\rho^k, \theta^k). \tag{3}$$

Corollary 3 *Let $E_{\inf} = \inf E(\rho, \theta) > -\infty$. For the sequence generated by Eq. 2, cumulative dissipation is strictly bounded by the initial energy gap: $\sum_{k=0}^{\infty} D(\rho^{k+1}, \rho^k) \leq h(E(\rho^0, \theta^0) - E_{\inf})$, and $\sum_{k=0}^{\infty} \|\theta^{k+1} - \theta^k\|_2^2 \leq \frac{2h}{\alpha}\big(E(\rho^0, \theta^0) - E_{\inf}\big)$. Consequently, $D(\rho^{k+1}, \rho^k) \to 0$ and $\|\theta^{k+1} - \theta^k\|_2 \to 0$ as $k \to \infty$.*

Remark 4 *Theorem 1 confirms the exact proximal scheme is well-posed: the compact simplex ensures minimizers exist, while Proposition 2 and Corollary 3 guarantee that topological "movement" is bounded and naturally vanishes over time, preventing the chaotic mask oscillations. While the exact proximal scheme guarantees strict monotonic descent, our practical implementation (Alg. 1) relies on stochastic gradients, meaning descent is realized in expectation.*

4.2. Practical implementation

To scale the exact full-batch proximal scheme (Eq. 2) to deep networks, we implement an inexact stochastic approximation. Rather than solving the proximal step analytically, we use an alternating inner-outer loop optimization.

Parameterization. We map the simplex allocation ρ to a continuous gate $M \in [0, 1]^{m \times n}$ using a temperature-scaled sigmoid: $M = \sigma\big(\tau \cdot (\log(\rho + \epsilon) - b)\big)$. The threshold b is dynamically computed per layer such that the mean gate value satisfies the target sparsity s, ensuring $\frac{1}{mn} \sum M = 1 - s$.

Inexact proximal optimization. At each outer step k, we fix the current parameters (ρ^k, θ^k) as anchor points. In the inner loop, we sample minibatches and perform T stochastic gradient steps to minimize the proximal objective: $\widehat{J}_k(\rho, \theta) = \mathcal{L}_{\text{task}} + \frac{1}{h} D(\rho, \rho^k) + \frac{\alpha}{2h} \|\theta - \theta^k\|_2^2$. We use exact gradients for smooth distances (e.g., L_2, KL) and subgradients or differentiable proxies for non-smooth ones (e.g., entropic regularizer for W_1). To strictly enforce the probability simplex constraint, we update ρ using the Cauchy-Simplex optimizer [7].

Evaluation. At test time, discrete binary subnetworks are extracted via a standard Top-K projection. While our theoretical guarantees (well-posedness and descent) apply strictly to the continuous variational dynamics prior to this non-differentiable projection, they justify our geometry-aware objective as a principled mechanism to stabilize allocation dynamics.

5. Experiments

5.1. Experimental setup

Training details. We evaluate our CSMM framework on CIFAR-10 using the ResNet-20. The allocation ρ is initialized uniformly, follow an Erdös-Reyni Kernel layer-wise sparsity budget. For our proximal dynamics, we run $K = 30$ outer steps, each containing $T = 5$ inner minibatch steps, with proximal hyperparameters $h = 1$ and $\alpha = 1$. Further implementation details are provided in Appendix B

Baseline methods. We compare our approach against several prominent continuous sparsification methods. *L1* utilizes standard magnitude-based pruning with iterative retraining, serving as a classic benchmark. *STR* [19] incorporates differentiable thresholding to learn sparsity during training. *spred* [39] reformulates sparse training as a LASSO-like optimization problem by jointly optimizing weights and continuous mask variables. Finally, *PiLoT* [16] builds on spred by adopting a mirror-flow framework that dynamically adapts weight decay, enabling a principled transition from L_2 to L_1 regularization. Unlike these methods, which apply unconstrained Euclidean updates to mask parameters, our framework explicitly regularizes the temporal evolution of connectivity on the simplex via geometry-aware proximal penalties.

5.2. Main results

Performance comparison. We compare our CSMM against established Continuous Sparsification baselines on ResNet-20/CIFAR-10 (Figure 1, left). CSMM consistently outperforms established baselines across 90%–97% sparsity range, demonstrating the effectiveness of its geometry-aware allocation dynamics. However, at extreme sparsity levels ($> 98\%$), CSMM's performance lags behind PiLoT. This is expected, as our current formulation relies on fixed layer-wise sparsity budgets (ERK), whereas PiLoT dynamically optimizes the sparsity distribution across layers during training.

Effect of distance metric across sparsity levels. Evaluating different proximal penalties (Figure 1, middle) reveals distinct scaling behaviors. While L_2, KL, and W_1 perform comparably up to 94% sparsity, W_1 degrades at 95%, whereas KL and L_2 remain highly stable. Conversely, L_1 consistently underperforms. This confirms that the choice of distance D is not a mere technical detail, but a critical inductive bias that directly shapes the accuracy-sparsity trade-off.

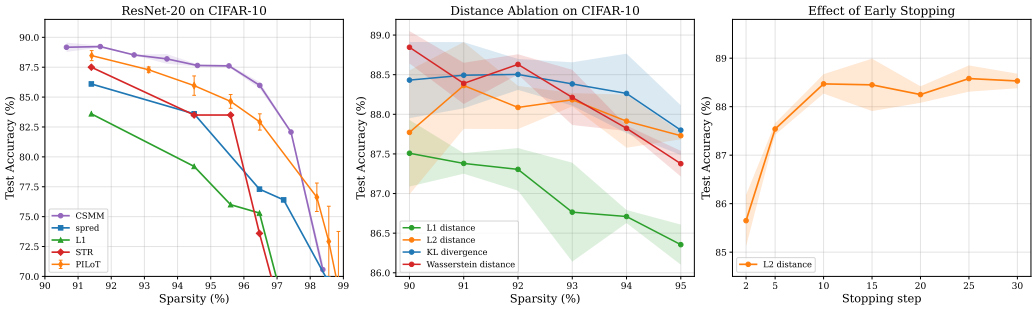

Figure 1: Performance analysis of CSMM on ResNet-20/CIFAR-10. (Left) Comparison against baselines across varying sparsity levels. (Middle) Ablation of the dissipation geometry D. (Right) Effect of early stopping the continuous graphon adaptation phase.

Effect of early stopping in graphon learning. We ablate the duration of continuous topology adaptation (90% sparsity) by freezing the allocation ρ after varying outer steps (Figure 1, right). While freezing too early hurts performance (e.g., 85.6% accuracy at step 2), performance saturates by just 10 outer steps. This shows that CSMM rapidly discovers high-quality topologies, allowing for efficient early stopping of the structural evolution phase without compromising final accuracy.

5.3. Linear mode connectivity of learned subnetworks

To test if CSMM identifies a coherent family of subnetworks rather than isolated optima, we analyze Linear Mode Connectivity (LMC) between pairs sampled from the final allocation. Compared to random and weight-reinitialized baselines, our subnetworks exhibit significantly flatter interpolation curves without pronounced loss barriers (Figure 2). This demonstrates that the learned allocation organizes a broad, shared low-loss basin. Furthermore, the steep loss barriers in the reinitialized model weight baseline confirm that this basin relies on the collaboration between the learned sparse topology and the trained weights. Please refer to Appendix B for details.

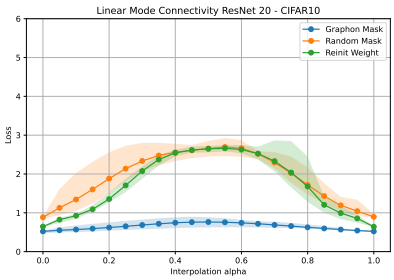

Figure 2: Linear mode connectivity

6. Conclusion

In this paper, we have presented a geometric framework for continuous sparsification, formulating connectivity evolution as a minimizing-movement scheme on the probability simplex. By decoupling the task objective from the geometry of connectivity reallocation, our method transforms sparse training into a controlled dynamical system. This allows practitioners to impose explicit inductive biases, ranging from Euclidean drift to transport-based structural constraints, through the choice of proximal penalty. Our empirical results on ResNet-20/CIFAR-10 suggest that this geometric perspective provides a promising and efficient approach to learning high-quality sparse subnetworks. Beyond performance gains, our framework offers a novel approach for future research into structural evolution in deep networks.

References

[1] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer, 2005.

[2] Christian Borgs, Jennifer T Chayes, László Lovász, Vera T Sós, and Katalin Vesztergombi. Convergent sequences of dense graphs i: Subgraph frequencies, metric properties and testing. *Advances in Mathematics*, 219(6):1801–1851, 2008.

[3] Leon Bungert, Tim Roith, Daniel Tenbrinck, and Martin Burger. A bregman learning framework for sparse neural networks. *Journal of Machine Learning Research*, 23(192):1–43, 2022.

[4] Rebekka Burkholz. Most activation functions can win the lottery without excessive depth. *Advances in Neural Information Processing Systems*, 35:18707–18720, 2022.

[5] Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. The lottery ticket hypothesis for pre-trained bert networks. *Advances in neural information processing systems*, 33:15834–15846, 2020.

[6] Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Michael Carbin, and Zhangyang Wang. The lottery tickets hypothesis for supervised and self-supervised pre-training in computer vision models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16306–16316, 2021.

[7] James Chok and Geoffrey M Vasil. Convex optimization over a probability simplex. *arXiv preprint arXiv:2305.09046*, 2023.

[8] Tim Dettmers and Luke Zettlemoyer. Sparse networks from scratch: Faster training without losing performance. *arXiv preprint arXiv:1907.04840*, 2019.

[9] Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the lottery: Making all tickets winners. In *International Conference on Machine Learning*, pages 2943–2952. PMLR, 2020.

[10] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

[11] Advait Gadhikar, Tom Jacobs, Chao Zhou, and Rebekka Burkholz. Sign-in to the lottery: Reparameterizing sparse training from scratch. *arXiv preprint arXiv:2504.12801*, 2025.

[12] Thomas Gebhart, Udit Saxena, and Paul Schrater. A unified paths perspective for pruning at initialization. *ArXiv*, abs/2101.10552, 2021.

[13] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

[14] Duc N.M Hoang, Shiwei Liu, Radu Marculescu, and Zhangyang Wang. REVISITING PRUNING AT INITIALIZATION THROUGH THE LENS OF RAMANUJAN GRAPH. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=uVcDssQff_.

[15] Torsten Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *J. Mach. Learn. Res.*, 22(1), jan 2021.

[16] Tom Jacobs and Rebekka Burkholz. Mask in the mirror: Implicit sparsification. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=U47ymTS3ut.

[17] Tom Jacobs, Advait Gadhikar, Celia Rubio-Madrigal, and Rebekka Burkholz. Hyperbolic aware minimization: Implicit bias for sparsity. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=XKB5Hu0ACY.

[18] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

[19] Aditya Kusupati, Vivek Ramanujan, Raghav Somani, Mitchell Wortsman, Prateek Jain, Sham Kakade, and Ali Farhadi. Soft threshold weight reparameterization for learnable sparsity. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5544–5555. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/kusupati20a.html.

[20] Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. *Advances in neural information processing systems*, 2, 1989.

[21] Namhoon Lee, Thalaiyasingam Ajanthan, and Philip Torr. Snip: Single-shot network pruning based on connection sensitivity. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1VZqjAcYX.

[22] Yunqiang Li, Jan C Van Gemert, Torsten Hoefler, Bert Moons, Evangelos Eleftheriou, and Bram-Ernst Verhoef. Differentiable transportation pruning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16957–16967, 2023.

[23] Shiwei Liu, Tianlong Chen, Xiaohan Chen, Zahra Atashgahi, Lu Yin, Huanyu Kou, Li Shen, Mykola Pechenizkiy, Zhangyang Wang, and Decebal Constantin Mocanu. Sparse training via boosting pruning plasticity with neuroregeneration. *Advances in Neural Information Processing Systems*, 34:9908–9922, 2021.

[24] Shiwei Liu, Lu Yin, Decebal Constantin Mocanu, and Mykola Pechenizkiy. Do we actually need dense over-parameterization? in-time over-parameterization in sparse training. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6989–7000. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/liu21y.html.

[25] Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l0 regularization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1Y8hhg0b.

[26] László Lovász and Balázs Szegedy. Limits of dense graph sequences. *Journal of Combinatorial Theory, Series B*, 96(6):933–957, 2006.

[27] Yannick Lunk, Sebastian J Scott, and Leon Bungert. Sparse training of neural networks based on multilevel mirror descent. *arXiv preprint arXiv:2602.03535*, 2026.

[28] Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H Nguyen, Madeleine Gibescu, and Antonio Liotta. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nature communications*, 9(1):1–12, 2018.

[29] Shreyas Malakarjun Patil and Constantine Dovrolis. Phew: Constructing sparse networks that learn fast and generalize well without training data. In *International Conference on Machine Learning*, pages 8432–8442. PMLR, 2021.

[30] Hoang Pham, Shiwei Liu, Lichuan Xiang, Dung Le, Hongkai Wen, Long Tran-Thanh, et al. Towards data-agnostic pruning at initialization: what makes a good sparse mask? *Advances in Neural Information Processing Systems*, 36:80044–80065, 2023.

[31] Hoang Pham, The-Anh Ta, Tom Jacobs, Rebekka Burkholz, and Long Tran-Thanh. The graphon limit hypothesis: Understanding neural network pruning via infinite width analysis. In *The Third Conference on Parsimony and Learning (Recent Spotlight Track)*, 2026. URL https://openreview.net/forum?id=HJkdDRmUzi.

[32] Hoang Pham, The-Anh Ta, and Long Tran-Thanh. Pruning at initialisation through the lens of graphon limit: Convergence, expressivity, and generalisation. *arXiv preprint arXiv:2602.06675*, 2026.

[33] Pedro Savarese, Hugo Silva, and Michael Maire. Winning the lottery with continuous sparsification. *Advances in neural information processing systems*, 33:11380–11390, 2020.

[34] Yucong Shen, Li Shen, Hao-Zhi Huang, Xuan Wang, and Wei Liu. Cpot: Channel pruning via optimal transport. *arXiv preprint arXiv:2005.10451*, 2020.

[35] Hidenori Tanaka, Daniel Kunin, Daniel L Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. *Advances in Neural Information Processing Systems*, 33:6377–6389, 2020.

[36] Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkgsACVKPH.

[37] Geng Yuan, Xiaolong Ma, Wei Niu, Zhengang Li, Zhenglun Kong, Ning Liu, Yifan Gong, Zheng Zhan, Chaoyang He, Qing Jin, et al. Mest: Accurate and fast memory-economic sparse training framework on the edge. *Advances in Neural Information Processing Systems*, 34: 20838–20850, 2021.

[38] Hattie Zhou, Janice Lan, Rosanne Liu, and Jason Yosinski. Deconstructing lottery tickets: Zeros, signs, and the supermask. *Advances in neural information processing systems*, 32, 2019.

[39] Liu Ziyin and Zihao Wang. spred: Solving l1 penalty with SGD. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 43407–43422. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/ziyin23a.html.

Appendix A. Proofs of theoretical results

A.1. Validation of energy regularity (Assumption 1)

In this section, we verify that the task energy function $E(\rho, \theta)$ formulated in our continuous sparsification framework naturally satisfies the conditions of Assumption 1.

Fix a finite training set $\mathcal{D} = \{(x_r, y_r)\}_{r=1}^N$. We assume the standard deep learning setup:

1. The soft-gate mapping $S : \Delta_{m,n} \to [0,1]^{m\times n}$ is continuous (e.g., our temperature-scaled sigmoid).
2. For each $r \in \{1, \dots, N\}$, the sample-wise network output map $(\rho, \theta) \mapsto f(x_r; \theta, S(\rho))$ is continuous.
3. For each r, the sample loss $u \mapsto \ell(u, y_r)$ is lower semicontinuous and bounded below (e.g., cross-entropy or mean-squared error).
4. The regularizers $R_\theta : \mathbb{R}^p \to (-\infty, +\infty]$ (e.g., L_2 weight decay) and $R_\rho : [0,1]^{m\times n} \to (-\infty, +\infty]$ (e.g., annealed entropy) are lower semicontinuous and bounded below.

We define the total task energy as the sum of the empirical loss and the regularizers:

$$E(\rho, \theta) := \frac{1}{N} \sum_{r=1}^{N} \ell\big(f(x_r; \theta, S(\rho)), y_r\big) + \lambda_\theta R_\theta(\theta) + \lambda_\rho R_\rho(S(\rho)), \tag{4}$$

where $\lambda_\theta, \lambda_\rho \geq 0$.

Lower semicontinuity. Let the sequence $(\rho^t, \theta^t) \to (\rho, \theta)$ in the product space $\Delta_{m,n} \times \mathbb{R}^p$. By the continuity of the gate mapping S and the network map f, we have:

$$f(x_r; \theta^t, S(\rho^t)) \to f(x_r; \theta, S(\rho)) \qquad \text{for each } r.$$

Because the sample loss $u \mapsto \ell(u, y_r)$ is lower semicontinuous, applying its definition yields:

$$\ell\big(f(x_r; \theta, S(\rho)), y_r\big) \leq \liminf_{t\to\infty} \ell\big(f(x_r; \theta^t, S(\rho^t)), y_r\big).$$

Similarly, by the lower semicontinuity of the regularizers R_θ and R_ρ, we obtain:

$$R_\theta(\theta) \leq \liminf_{t\to\infty} R_\theta(\theta^t), \qquad R_\rho(S(\rho)) \leq \liminf_{t\to\infty} R_\rho(S(\rho^t)).$$

Applying the elementary inequality for limits of sequences, $\liminf_{t\to\infty}(a_t + b_t) \geq \liminf_{t\to\infty} a_t + \liminf_{t\to\infty} b_t$, repeatedly over the finitely many non-negative terms in Eq. (4), we conclude that:

$$E(\rho, \theta) \leq \liminf_{t\to\infty} E(\rho^t, \theta^t).$$

Hence, the total energy E is lower semicontinuous.

Boundedness from below. Next, we show that E is bounded below. Since each sample loss and each regularizer is bounded below by construction, there exist constants $c_\ell, c_\theta, c_\rho \in \mathbb{R}$ such that:

$$\ell(u, y_r) \geq c_\ell, \qquad R_\theta(\theta) \geq c_\theta, \qquad R_\rho(m) \geq c_\rho$$

for all admissible arguments. Therefore, factoring in the non-negative hyperparameter weights, we have:

$$E(\rho, \theta) \geq c_\ell + \lambda_\theta c_\theta + \lambda_\rho c_\rho \qquad \text{for all } (\rho, \theta) \in \Delta_{m,n} \times \mathbb{R}^p.$$

Because E is both bounded below and lower semicontinuous, Assumption 1 is formally satisfied. This verification confirms that our exact proximal optimization theory safely encompasses the standard soft-gated architectures used in our practical implementation, ensuring that the analytical guarantees naturally extend to standard neural network layers.

A.2. Proof of Theorem 1

For fixed k, define the proximal objective

$$J_k(\rho, \theta) := E(\rho, \theta) + \frac{1}{h} D(\rho, \rho^k) + \frac{\alpha}{2h} \|\theta - \theta^k\|_2^2 \tag{5}$$

on $\Delta_{m,n} \times \mathbb{R}^p$.

We prove that J_k attains its infimum by the direct method of the calculus of variations.

J_k is bounded below and admits a minimizing sequence. By Assumption 1, the energy E is bounded below; write

$$E_{\inf} := \inf_{(\rho,\theta) \in \Delta_{m,n} \times \mathbb{R}^p} E(\rho, \theta) > -\infty. \tag{6}$$

By Assumption 2, $D(\rho, \rho^k) \geq 0$ for all ρ, and the quadratic term is also nonnegative. Hence

$$J_k(\rho, \theta) \geq E_{\inf} > -\infty \qquad \text{for all } (\rho, \theta). \tag{7}$$

Therefore $\inf J_k > -\infty$, and there exists a minimizing sequence $\{(\rho^t, \theta^t)\}_{t \geq 1}$ such that

$$J_k(\rho^t, \theta^t) \to \inf J_k \qquad \text{as } t \to \infty. \tag{8}$$

Compactness in ρ and boundedness in θ. Since $\Delta_{m,n}$ is a closed and bounded subset of the finite-dimensional space $\mathbb{R}^{m \times n}$, it is compact. Hence, after passing to a subsequence if necessary, we may assume

$$\rho^t \to \bar{\rho} \in \Delta_{m,n}. \tag{9}$$

We now show that $\{\theta^t\}$ is bounded. Since (ρ^k, θ^k) is an admissible competitor and $E(\rho^k, \theta^k) < \infty$, we have

$$J_k(\rho^k, \theta^k) = E(\rho^k, \theta^k) + \frac{1}{h} D(\rho^k, \rho^k) + \frac{\alpha}{2h} \|\theta^k - \theta^k\|_2^2 = E(\rho^k, \theta^k), \tag{10}$$

where we used Assumption 2. Since (ρ^t, θ^t) is minimizing, there exists t_0 such that for all $t \geq t_0$,

$$J_k(\rho^t, \theta^t) \leq \inf J_k + 1 \leq J_k(\rho^k, \theta^k) + 1 = E(\rho^k, \theta^k) + 1. \tag{11}$$

Using $E(\rho^t, \theta^t) \geq E_{\inf}$ and $D(\rho^t, \rho^k) \geq 0$, we get

$$\frac{\alpha}{2h}\|\theta^t - \theta^k\|_2^2 \leq J_k(\rho^t, \theta^t) - E(\rho^t, \theta^t) \leq E(\rho^k, \theta^k) + 1 - E_{\inf}. \tag{12}$$

Because $\alpha > 0$, this yields a uniform bound on $\|\theta^t - \theta^k\|_2$, hence on $\|\theta^t\|_2$. Thus $\{\theta^t\}$ is bounded in $\mathbb{R}^p$. By Bolzano–Weierstrass, after passing to another subsequence if necessary, we may assume

$$\theta^t \to \bar{\theta} \in \mathbb{R}^p. \tag{13}$$

Lower semicontinuity gives attainment of the infimum. By Assumption 1,

$$E(\bar{\rho}, \bar{\theta}) \leq \liminf_{t\to\infty} E(\rho^t, \theta^t). \tag{14}$$

By Assumption 2, the map $\rho \mapsto D(\rho, \rho^k)$ is lower semicontinuous, so

$$D(\bar{\rho}, \rho^k) \leq \liminf_{t\to\infty} D(\rho^t, \rho^k). \tag{15}$$

The map $\theta \mapsto \|\theta - \theta^k\|_2^2$ is continuous, hence

$$\|\bar{\theta} - \theta^k\|_2^2 = \lim_{t\to\infty} \|\theta^t - \theta^k\|_2^2. \tag{16}$$

Combining these and using again the elementary inequality for $\liminf$ of sums, we obtain

$$J_k(\bar{\rho}, \bar{\theta}) \leq \liminf_{t\to\infty} J_k(\rho^t, \theta^t) = \inf J_k. \tag{17}$$

Therefore $(\bar{\rho}, \bar{\theta})$ attains the infimum of J_k, and hence is a minimizer of Eq. 2. This proves the theorem. □

A.3. Proof of Proposition 2

By optimality of $(\rho^{k+1}, \theta^{k+1})$ for Eq. 2, we have

$$J_k(\rho^{k+1}, \theta^{k+1}) \leq J_k(\rho^k, \theta^k). \tag{18}$$

By Assumption 2,

$$D(\rho^k, \rho^k) = 0, \tag{19}$$

and clearly

$$\|\theta^k - \theta^k\|_2^2 = 0. \tag{20}$$

Therefore

$$J_k(\rho^k, \theta^k) = E(\rho^k, \theta^k). \tag{21}$$

Expanding the definition of J_k at $(\rho^{k+1}, \theta^{k+1})$ yields

$$E(\rho^{k+1}, \theta^{k+1}) + \frac{1}{h}D(\rho^{k+1}, \rho^k) + \frac{\alpha}{2h}\|\theta^{k+1} - \theta^k\|_2^2 \leq E(\rho^k, \theta^k), \tag{22}$$

which is exactly Eq. 3. □

A.4. Proof of Corollary 3

Rearranging Eq. 3 gives, for every $k \geq 0$,

$$\frac{1}{h} D(\rho^{k+1}, \rho^k) + \frac{\alpha}{2h} \|\theta^{k+1} - \theta^k\|_2^2 \leq E(\rho^k, \theta^k) - E(\rho^{k+1}, \theta^{k+1}). \tag{23}$$

Summing from $k = 0$ to $N - 1$ yields

$$\sum_{k=0}^{N-1} \frac{1}{h} D(\rho^{k+1}, \rho^k) + \sum_{k=0}^{N-1} \frac{\alpha}{2h} \|\theta^{k+1} - \theta^k\|_2^2 \leq \sum_{k=0}^{N-1} \Big(E(\rho^k, \theta^k) - E(\rho^{k+1}, \theta^{k+1}) \Big). \tag{24}$$

The right-hand side telescopes:

$$\sum_{k=0}^{N-1} \Big(E(\rho^k, \theta^k) - E(\rho^{k+1}, \theta^{k+1}) \Big) = E(\rho^0, \theta^0) - E(\rho^N, \theta^N). \tag{25}$$

Since $E(\rho^N, \theta^N) \geq E_{\text{inf}}$, we obtain

$$\sum_{k=0}^{N-1} \frac{1}{h} D(\rho^{k+1}, \rho^k) + \sum_{k=0}^{N-1} \frac{\alpha}{2h} \|\theta^{k+1} - \theta^k\|_2^2 \leq E(\rho^0, \theta^0) - E_{\text{inf}}. \tag{26}$$

This gives

$$\sum_{k=0}^{N-1} D(\rho^{k+1}, \rho^k) \leq h(E(\rho^0, \theta^0) - E_{\text{inf}}) \tag{27}$$

$$\sum_{k=0}^{N-1} \|\theta^{k+1} - \theta^k\|_2^2 \leq \frac{2h}{\alpha} (E(\rho^0, \theta^0) - E_{\text{inf}}) \tag{28}$$

It remains to show that the increments vanish. Both sequences

$$a_k := D(\rho^{k+1}, \rho^k) \geq 0, \qquad b_k := \|\theta^{k+1} - \theta^k\|_2^2 \geq 0 \tag{29}$$

are nonnegative, and bounds in Eq. 27 and Eq. 28 show that

$$\sum_{k=0}^{\infty} a_k < \infty, \qquad \sum_{k=0}^{\infty} b_k < \infty. \tag{30}$$

Every convergent series with nonnegative terms has terms tending to zero, hence

$$D(\rho^{k+1}, \rho^k) \to 0, \qquad \|\theta^{k+1} - \theta^k\|_2^2 \to 0. \tag{31}$$

Taking square roots in the second relation gives

$$\|\theta^{k+1} - \theta^k\|_2 \to 0. \tag{32}$$

This proves the corollary. □

A.5. Remark on the multi-layer case

For a finite family of masked layers $\ell \in \mathcal{L}$, let

$$\rho = (\rho_\ell)_{\ell \in \mathcal{L}} \in \prod_{\ell \in \mathcal{L}} \Delta_\ell. \tag{33}$$

Assume the energy is lower semicontinuous and bounded below on the product space, and that each layerwise dissipation $D_\ell(\cdot, \rho_\ell^k)$ is nonnegative, lower semicontinuous, and vanishes on the diagonal. Define

$$J_k(\rho, \theta) = E(\rho, \theta) + \frac{1}{h} \sum_{\ell \in \mathcal{L}} D_\ell(\rho_\ell, \rho_\ell^k) + \frac{\alpha}{2h} \|\theta - \theta^k\|_2^2. \tag{34}$$

Because a finite product of compact simplices is compact, the proof of Theorem 1 applies verbatim. The descent and summability statements also follow by the same argument, with $\sum_{\ell \in \mathcal{L}} D_\ell(\rho_\ell^{k+1}, \rho_\ell^k)$ replacing the single-layer term.

Appendix B. Experimental details

For the ResNet-20/CIFAR-10 experiments, we train on CIFAR-10 using standard data augmentation consisting of random crops and random horizontal flips, followed by normalization with dataset-specific channel statistics. Unless otherwise stated, we use a batch size of 256 and optimize the network weights with SGD with Nesterov momentum 0.9, learning rate 0.1, weight decay 10^{-4}, and the default cosine annealing schedule. The ρ is uniformly initialized, we use ERK-based layerwise sparsity initialization. We leverage global Cauchy-Simplex optimizer [7] with simplex learning rate 0.1 to enforce that ρ be within simplex. We run 30 outer minimizing-movement steps, each with 5 inner optimization steps, using $h = 1$, $\alpha = 1$, $\lambda_\theta = 10^{-4}$, and entropy regularization $\lambda_\rho = 0.1$; hard subnetworks are obtained by Top-K projection at the target sparsity level. For the distance-ablation experiments, we evaluate target sparsities from 90% to 95% and freeze the graphon after step 20 unless that freezing time is itself the variable under study.

Algorithm 1 gives the practical stochastic realization of the exact proximal update analyzed in Section 4. The inner loop approximates the proximal minimizer using minibatch gradients and simplex projection.

Linear mode connectivity (LMC) details. To evaluate the LMC of our learned topology on ResNet-20/CIFAR-10, we utilize the model checkpoint obtained after 30 minimizing-movement training steps at a 90% target sparsity. From this trained continuous allocation ρ, we sample pairs of hard binary masks, M_1^{binary} and M_2^{binary}. The corresponding endpoint subnetworks are fine-tuned for 20 epochs using a learning rate of 10^{-3}. For the weight-reinitialized baseline, we apply the exact same graphon-sampled masks, but reinitialize the underlying weights and fine-tune them from scratch for 100 epochs to ensure convergence. As a secondary baseline, we replace the graphon-derived masks with uniformly random binary masks at the same 90% sparsity. To trace the connectivity path, we linearly interpolate the masked weights over 21 evenly points $\gamma \in [0, 1]$. Specifically, the intermediate weights are computed as $\theta_\alpha = (1 - \gamma)(\theta_1 \odot M_1^{\text{binary}}) + \gamma(\theta_2 \odot M_2^{\text{binary}})$. We note that because the two sampled masks are not identical, the intermediate models for $\alpha \in (0, 1)$ possess active connections representing the union of the two masks ($M_1 \cup M_2$), resulting in a slightly lower sparsity along the interpolation path. All LMC evaluations are averaged over 5 independent runs.

Algorithm 1 Minimizing Movement Sparse Training

Input: training data $\mathcal{D}_{\text{train}}$, network f_θ, masked layers $\mathcal{L}$,
initial weights θ^0, initial allocation $\rho^0 \in \Delta$,
gate map S, dissipation D,
outer steps K, inner steps T, step size h, proximal weight α,
learning rates η_θ, η_ρ, target sparsity s
Output: Sparse network f_{θ^K} with mask $M^\star_{binary}$
$E(\rho, \theta; \mathcal{D}_{\text{train}}) = \mathcal{L}_{\text{task}}(\theta, \rho; \mathcal{D}_{\text{train}}) + \lambda_\theta R_\theta(\theta) + \lambda_\rho R_\rho(S(\rho))$ # Define energy function
for $k \leftarrow 0$ **to** $K-1$ **do**
  $\bar{\theta} \leftarrow \theta^k, \quad \bar{\rho} \leftarrow \rho^k$
  $\theta \leftarrow \theta^k, \quad \rho \leftarrow \rho^k$
  for $t \leftarrow 1$ **to** T **do**
    Sample minibatch $B \subset \mathcal{D}_{\text{train}}$
    $m \leftarrow S(\rho)$
    $\widehat{J}_k(\rho, \theta; B) = E(\rho, \theta; B) + \frac{1}{h} D(\rho, \bar{\rho}) + \frac{\alpha}{2h} \|\theta - \bar{\theta}\|_2^2$ # Define stochastic proximal objective
    $\theta \leftarrow \theta - \eta_\theta \nabla_\theta \widehat{J}_k(\rho, \theta; B)$ # Update weights
    $\rho \leftarrow \rho - \eta_\rho \nabla_\rho \widehat{J}_k(\rho, \theta; B)$ # Update allocation
    $\rho \leftarrow \Pi_\Delta(\rho)$ # Project onto simplex
  end
  $\theta^{k+1} \leftarrow \theta \;\; \rho^{k+1} \leftarrow \rho$
end
$M^\star \leftarrow S(\rho^K)$
$M^\star_{binary} \leftarrow \text{TopK}(m^\star, 1-s)$
return f_{θ^K} *with mask* $M^\star_{binary}$