SPARSITY WITHOUT SWEEPS: ADAPTIVE REGULARIZATION FOR MIRROR DESCENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Sparse training reduces the memory and computational costs of deep neural networks. However, sparse optimization methods, e.g., those adding an ℓ_1 penalty, often control sparsity only indirectly through a regularization parameter λ, whose mapping to the final sparsity rate is highly non-trivial. This work addresses the selection of λ in the mirror descent (MD) sparse optimization framework such that a user-specified sparsity rate is achieved. Instead of treating this selection as a meta-optimization problem, which potentially requires expensive trial-and-error sweeps, our approach adapts λ online during training. We propose two schemes to perform the adaptation: one based on the magnitude of the largest subgradients, and the other based on the difference between the model's current and target sparsity. The first scheme can be interpreted as a dynamic sparse training cycle. A sparsity schedule further extends this control to intermediate targets, gradually sparsifying the model during training. We also provide a convergence analysis for adaptive λ in MD. Experiments on CIFAR-100, TinyImageNet, and ImageNet-1K show that both schemes reliably reach the final as well as the scheduled intermediate sparsity targets. Beyond controlling sparsity, the adaptive schemes match or exceed the accuracy of a tuned fixed λ at equal or higher sparsity in every evaluated setting, and clearly surpass it when combined with a sparsity schedule: with ResNet-50 on TinyImageNet, they reach 67.1% top-1 accuracy at 95% sparsity, compared with 64.6% for a fixed λ at only 90.4% sparsity.

1 INTRODUCTION

1.1 BACKGROUND

The success of deep neural networks (DNNs) across a wide range of applications has been accompanied by an exponential increase in scale (Sevilla et al., 2022). This has led to growing concern regarding the environmental cost of scaling DNNs (Dhar, 2020; Luccioni et al., 2023; Strubell et al., 2019). In response, a significant body of research is devoted to developing more efficient models. One such approach is learning sparse models, or *sparsification*, which is comprehensively reviewed by Hoefler et al. (2021). Focusing on weight sparsity, we divide existing methods into three broad categories: *pruning*, *dynamic sparse training*, and *sparse optimization*.

Pruning removes connections from the network typically based on their magnitude. Some pruning methods apply pruning gradually during training based on sparsity schedules (Zhu & Gupta, 2017), or post-training followed by a fine-tuning step (Han et al., 2015). Other methods apply a fixed pruning mask at initialization, resulting in sparse forward and backward passes. This includes SNIP (Lee et al., 2019) and SynFlow (Tanaka et al., 2020), which utilize first-order gradient information as a pruning criterion. GraSP (Wang et al., 2020) and FISH (Sung et al., 2021) use second-order metrics based on the Hessian and Fisher information, respectively.

Dynamic sparse training (DST) follows *train-prune-grow* cycles to dynamically adjust the sparse network topology. SET (Mocanu et al., 2018) applies magnitude-based pruning while regrowing connections at random, allowing for sparse gradients. RigL (Evci et al., 2020) replaces the random regrowth by gradient magnitude, whereas Dettmers & Zettlemoyer (2019) uses a momentum growth criterion. Top-KAST (Jayakumar et al., 2020) maintains a sparse backward pass over a subset of weights larger than the active topology to balance exploration and efficiency. GraNet (Liu et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2021a) pairs RigL's growth criterion with global pruning and a progressively increasing sparsity instead of fixed layer-wise sparsity. NeurRev (Li et al., 2024) refines the pruning criterion by additionally removing the least-changing negative weights to revitalize dormant neurons. BiDST (Ji et al., 2024) jointly optimizes the weights and pruning mask through a bilevel formulation.

Sparse optimization takes a different approach by incorporating sparsity-inducing penalties into the training objective. A common approach is to add an explicit ℓ_1 regularization term (Tibshirani, 1996), which serves as a convex relaxation of the ℓ_0 penalty. Louizos et al. (2018) presented a differentiable ℓ_0-regularization via stochastic gates on the weights. Other approaches use *implicit* regularization, rather than an explicit penalty term, which can be achieved using mirror descent (MD) (Nemirovsky & Yudin, 1983). A series of works (Huang et al., 2016; Azizan et al., 2022; Bungert et al., 2022; Wang & Benning, 2023; Lunk et al., 2026) leverage MD and *linearized Bregman iterations* to induce sparsity without explicit regularization.

1.2 SPARSITY CONTROL IN SPARSE OPTIMIZATION FRAMEWORKS

A key challenge in sparse optimization is to steer the sparsity rate of the trained model exactly to a prescribed target. Sparse optimization methods typically control sparsity only indirectly via a regularization parameter λ, whose mapping to a given sparsity rate is highly non-trivial.

In practice, the desired sparsity is often obtained by sweeping over λ, turning sparsity control into a meta-optimization that requires a full training run per candidate value (Gale et al., 2019; Bungert et al., 2022; Kusupati et al., 2020). This also applies to principled search approaches, such as the iterative selection algorithm of Shen et al. (2024). Adaptive schemes avoid this meta-optimization altogether by updating λ within a single training run, typically based on the gap between the current and the target sparsity. Kim et al. (2016) adjusted per-layer ℓ_1 coefficients for fully-connected networks with predefined per-layer targets. Yuan et al. (2021) scaled the penalty coefficients by the sparsity gap, whereas Gallego-Posada et al. (2022) controlled λ such that the density of the gates of Louizos et al. (2018) reaches the target. The discussed schemes adapt an explicit penalty or constraint in the objective, and do not directly extend to the implicit regularization in MD, where λ enters through the functional defining the Bregman divergence rather than the objective function.

1.3 CONTRIBUTIONS

In this paper, we introduce sparsity control for MD, specifically the setup in Bungert et al. (2022), and propose two adaptation schemes for λ to drive sparsity towards a target in a single run. We validate our method on three common image classification benchmarks, namely CIFAR-100, TinyImageNet, and ImageNet-1K. Our main contributions are summarized as follows:

(I) We extend Bungert et al. (2022) by incorporating an adaptive regularization parameter λ to train neural networks with a specific *target sparsity* as well as multiple intermediate targets set by a predefined sparsity schedule. To adapt λ, we propose two schemes: TopK, which sets λ based on the largest subgradient magnitudes, and SGap, which updates λ based on the gap between the current and target sparsity. We also provide a convergence analysis for adaptive λ, building upon the technique developed in Benning et al. (2021).

(II) We interpret the MD iterates, specifically linearized Bregman (LinBreg), with largest subgradients adaptation through the lens of DST. Under some conditions, we show that the adaptive LinBreg iterates perform a DST-like training cycle with magnitude-based pruning and accumulated gradient growing criteria.

(III) Beyond sparsity control, we empirically show that models trained with our proposed schemes outperform those trained with a fixed λ at comparable sparsity levels.

2 SPARSITY-CONSTRAINED MIRROR DESCENT

The goal in this work is the training of neural networks under a sparsity constraint. Particularly, for some supervised dataset $\mathcal{T}$ we minimize the loss function $\mathcal{L}(\theta) := \sum_{(x,y)\in\mathcal{T}} \ell\big(F(x;\theta),\, y\big)$, where $\theta \in \Theta \subset \mathbb{R}^d$ are the d-dimensional parameters of a neural network $F(\cdot;\theta)$ and $\ell(\cdot,\cdot)$ denotes the per-sample loss. To compute a sparse solution, we prescribe a target number of non-zero parameters

$\mathsf{k}^* \in \mathbb{N}$ and solve

$$\min_{\theta:|\theta|_0 \leq \mathsf{k}^*} \mathcal{L}(\theta), \tag{1}$$

where $|\cdot|_0$ denotes the ℓ_0-functional on $\mathbb{R}^d$. This is connected to the sparsity, which we define as

$$\mathsf{s}^* := \frac{d - \mathsf{k}^*}{d}, \qquad \mathsf{s}(\theta) := \frac{d - |\theta|_0}{d}. \tag{2}$$

Remark 1. *In the context of neural networks, the* Lottery Ticket Hypothesis *(Frankle & Carbin, 2019) suggests the existence of sparse sub-networks that match the performance of their dense counterparts. In our notation, this translates to the existence of* $\mathsf{k}^* < d$ *s.t.* $\min_{\theta:|\theta|_0 \leq \mathsf{k}^*} \mathcal{L}(\theta) = \min_{\theta \in \Theta} \mathcal{L}(\theta)$.

In order to solve (1), we employ an iterative scheme based on the inverse scale space approaches introduced in (Bungert et al., 2022; Osher et al., 2005), i.e., we start with an initialization $\theta^{(0)} \in \mathbb{R}^d$ with very few active components and in each iteration t desire the following properties:

(I1) $\mathcal{L}(\theta^{(t+1)}) < \mathcal{L}(\theta^{(t)})$: loss decay,
(I2) $|\theta^{(t+1)}|_0 \approx |\theta^{(t)}|_0$: sparsity preservation,
} MD framework (Bungert et al., 2022)

(I3) $|\theta^{(t+1)}|_0 \approx \mathsf{k}^*$: sparsity constraint. } Our contribution

We address (I1) and (I2) by defining the next iterate to be the solution of the following problem,

$$\theta^{(t+1)} = \arg\min_{\theta \in \Theta} \langle \nabla \mathcal{L}(\theta^{(t)}), \theta - \theta^{(t)} \rangle + \frac{1}{\tau} \mathcal{R}(\theta, \theta^{(t)}), \tag{3}$$

where we linearize $\mathcal{L}$ around the previous iterate, i.e., we replace $\mathcal{L}(\theta)$ by $\langle \nabla \mathcal{L}(\theta^{(t)}), \theta \rangle$. The functional $\mathcal{R} : \Theta^2 \to \mathbb{R} \cup \{+\infty\}$ promotes (I2), where its strength is weighted by the parameter $\tau > 0$. While this can be handled by the framework in (Bungert et al., 2022), it does not yield any control over the actual sparsity of $\theta^{(t+1)}$, only over its increment compared to $\theta^{(t)}$. The main contribution of this paper is to address (I3), i.e., keeping the iterates within a desired sparsity regime. This can be achieved through a suitable adaptation strategy for $\mathcal{R}$, as described in the following.

2.1 MIRROR DESCENT AND LINEARIZED BREGMAN ITERATIONS

The core element of (3) is the regularization functional $\mathcal{R}$. For example $\mathcal{R}(\theta, \tilde{\theta}) = 1/2|\theta - \tilde{\theta}|_2^2$ yields a standard gradient descent scheme. A popular choice is the *Bregman divergence* (cf. Bauschke & Combettes, 2017; Rockafellar & Wets, 1998)

$$D_\phi^p(\theta, \tilde{\theta}) := \phi(\theta) - \phi(\tilde{\theta}) - \langle p, \theta - \tilde{\theta} \rangle, \qquad p \in \partial\phi(\tilde{\theta}),$$

where $\partial\phi$ denotes the subdifferential of the convex functional $\phi : \Theta \to \mathbb{R}$. In particular, setting $\mathcal{R}(\theta, \theta^{(t)}) = D_\phi^{p^{(t)}}(\theta, \theta^{(t)})$ with $p^{(t)} \in \partial\phi(\theta^{(t)})$ in (3) leads to a family of methods known in the literature as *mirror descent*, which we can write as

$$\theta^{(t+1)} = \arg\min_{\theta \in \mathbb{R}^d} \left\{ \langle \theta - \theta^{(t)}, \nabla \mathcal{L}(\theta^{(t)}) \rangle + \frac{1}{\tau} D_\phi^{p^{(t)}}(\theta, \theta^{(t)}) \right\} = \nabla\phi^* \left(p^{(t)} - \tau \nabla \mathcal{L}(\theta^{(t)}) \right), \tag{4}$$

where $\nabla\phi^* = (\partial\phi)^{-1}$ is the gradient of the convex conjugate ϕ^* of ϕ. Applying $\partial\phi$ to both sides in (4), we observe that setting $p^{(t+1)} = p^{(t)} - \tau \nabla \mathcal{L}(\theta^{(t)}) \in \partial\phi(\theta^{(t+1)})$ yields a valid sequence of subgradients. This is known as *lazy mirror descent*, Nesterov's dual averaging (DA) (Nesterov, 2009), or LinBreg iterations (Yin et al., 2008; Cai et al., 2009). Furthermore, by replacing the simple gradient step with Adam-style momentum (Kingma & Ba, 2015), one obtains *AdaBreg* as introduced in (Bungert et al., 2022). For the choice of ϕ, we replace the ℓ_0-functional by its convex surrogate $\lambda|\theta|_1$ and add a quadratic term to ensure strong convexity, arriving at the elastic net functional $\phi_{\delta,\lambda}(\theta) := 1/2\delta|\theta|_2^2 + \lambda|\theta|_1$. The mirror map $\nabla\phi^*$ is then defined in closed-form as an element-wise soft-threshold (Bungert et al., 2022),

$$\theta^{(t+1)} = \nabla\phi^*(p^{(t+1)}) = \operatorname{prox}_{\delta\lambda|\cdot|_1}(\delta p^{(t+1)}) = \delta \left(\operatorname{sign}(p_i^{(t+1)}) \max\{|p_i^{(t+1)}| - \lambda, 0\}_{i=1}^d \right), \tag{5}$$

where $\delta > 0$, and $\operatorname{prox}_J(p) := \arg\min_w \frac{1}{2}|p - w|_2^2 + J(w)$ denotes the proximal operator. To simplify the notation, we assume $\delta = 1$ and parameterize $\phi_{\delta,\lambda}(\theta)$ as $\phi_\lambda(\theta)$ in the following.

2.2 PROMOTING THE TARGET SPARSITY VIA PARAMETER ADAPTATION

In the classical setting of LinBreg one considers a loss function $\mathcal{L}(\theta) = \ell(A\,\theta, y)$ for some linear operator A and function ℓ. One can show that with suitable step sizes the iterates converge to the closest minimizer in Bregman distance, that is $\theta^{(t)} \xrightarrow{t\to\infty} \arg\min_{\theta:\mathcal{L}(\theta)=0} D_\phi^{p^{(0)}}(\theta, \theta^{(0)})$, see (Gunasekar et al., 2018; Zhang et al., 2010; Yin et al., 2008). While this illustrates the algorithm's implicit bias, we cannot expect a similar result to hold in our case. In general, the dependence of the limit of $\theta^{(t)}$ as $t \to \infty$ on the parameter λ is hard to characterize, and there is no explicit selection rule for λ available. Instead, we adapt it iteratively to ensure that (I3) is fulfilled. Namely, for a sequence of parameters $\lambda^{(t)} \geq 0$, (5) is modified to

$$p^{(t+1)} = p^{(t)} - \tau\,\nabla\mathcal{L}\left(\theta^{(t)}\right), \qquad \theta^{(t+1)} = \nabla\phi^*_{\lambda^{(t+1)}}\left(p^{(t+1)}\right). \tag{6}$$

This modification introduces a tension in the setup of (5), in the sense that $p^{(t)} \in \partial\phi_{\lambda^{(t)}}(\theta^{(t)})$ but possibly $p^{(t)} \notin \partial\phi_{\lambda^{(t+1)}}(\theta^{(t)})$.

Remark 2. *Even for adaptive $\phi_{\lambda^{(t)}}$, it is possible to apply a subgradient correction scheme, which projects $p^{(t+1)}$ onto $\partial\phi_{\lambda^{(t+1)}}(\theta^{(t+1)})$ so that every step is again an exact mirror descent step. We propose a method for subgradient correction in Appendix B. However, we empirically found that such a scheme performs slightly worse than the simpler update in* (6)*, as reported in Appendix E.2.*

2.3 CONVERGENCE ANALYSIS

Our scheme falls outside the standard analysis of linearized Bregman iterations due to the violated subgradient property. We can still analyze the theoretical properties, which we do in this section and refer to Benfenati & Ruggiero (2013) for a related convergence analysis of inexact Bregman methods. The proofs and technical details are deferred to the appendix. We want to highlight two shortcomings of our analysis that we leave for future work: First, we work with the true gradients instead of stochastic estimators as done in Bungert et al. (2022). Furthermore, we assume the parameter sequence $(\lambda^{(t)})_{t\in\mathbb{N}}$ to be given and do not take into account the feedback loop introduced in our main algorithm, where $\lambda^{(t)}$ also depends on the iterate $\theta^{(t-1)}$ itself. Throughout our analysis we use the following assumption:

Assumption 1. *We have $\lambda^{(t)} > 0$, $\forall t \in \mathbb{N}$ and $\mathcal{L}$ is an L-smooth function bounded from below by 0 with bounded sublevel sets.*

We first state a loss decay up to an error that can be controlled by the magnitude of the change in $\lambda^{(t)}$.

Lemma 1. *Using assumption 1 we obtain*

$$\mathcal{L}(\theta^{(t+1)}) + \left(\frac{1}{\tau} - \frac{L}{2}\right)\left|\theta^{(t+1)} - \theta^{(t)}\right|_2^2 + \frac{\lambda^{(t+1)} - \lambda^{(t)}}{\tau}\left(\left|\theta^{(t+1)}\right|_1 - \left|\theta^{(t)}\right|_1\right) \leq \mathcal{L}(\theta^{(t)}).$$

In order to obtain some form of convergence result, we need to assume that the change in $\lambda^{(t)}$ decays fast enough. In the following we use the notation $\ell^p(\mathbb{R})$ to denote the space of sequences, such that $(|a^{(t)}|^p)_{t\in\mathbb{N}}$ is summable for all $(a^{(t)})_{t\in\mathbb{N}} \in \ell^p(\mathbb{R})$. With this we have the following lemma.

Lemma 2. *Let assumption 1 hold true, $\tau < \frac{2}{L}$ and let $(\lambda^{(t+1)} - \lambda^{(t)})_{t\in\mathbb{N}} \in \ell^2(\mathbb{R})$, then the sequence $\theta^{(t)}$ is bounded and furthermore $\sum_{t=0}^{\infty}\left|\theta^{(t+1)} - \theta^{(t)}\right|_2^2 < \infty$.*

This only yields convergence up to a subsequence. To obtain convergence of the full sequence, we need to assume that $(\lambda^{(t)})_{t\in\mathbb{N}}$ converges quickly enough to some limit $\lambda^{(\infty)} < \infty$. Our proof strategy here follows along the lines of Benning et al. (2021) with an additional error term introduced due the subgradient mismatch. We also need to assume a Kurdyka–Łojasiewicz condition on a surrogate functional, which we give in assumption 2. This can for example be ensured if $\mathcal{L}$ is semi-algebraic.

Proposition 1. *Let assumption 1 and 2 be fulfilled and furthermore assume that $|\lambda^{(t)} - \lambda^{(\infty)}| \leq C_\lambda t^q$ with $C_\lambda > 0, q < -3/2$ and $\tau < \frac{2}{L}$ then we have that*

$$\lim_{t\to\infty}\theta^{(t)} = \theta^{(\infty)} \qquad \textit{and} \qquad \nabla\mathcal{L}(\theta^{(\infty)}) = 0.$$

2.4 PROPOSED ADAPTATION STRATEGIES

Due to the integer nature of sparsity and stochastic optimization, it might be infeasible to strictly enforce $\mathsf{s}(\theta) = \mathsf{s}^*$ at every iteration. Instead, for a small tolerance ζ, we require the converged model θ^* to satisfy the relaxed constraint

$$\frac{|\mathsf{s}(\theta^*) - \mathsf{s}^*|}{\mathsf{s}^*} \leq \zeta, \qquad 1 \gg \zeta > 0. \tag{7}$$

Sparsity is controlled by adapting $\lambda^{(t)}$ in fixed intervals of length $\Delta t \in \mathbb{N}$, where $\lambda^{(t)}$ is updated once every Δt iterations and kept fixed in the remaining iterations. We refer to the former as *adaptation step* and to the latter as *freezing steps*. In the following, we describe our proposed adaptation schemes.

2.4.1 TOPK SUBGRADIENTS ADAPTATION

Let $|p^{(t)}|_{[1]} \leq |p^{(t)}|_{[2]} \leq \ldots \leq |p^{(t)}|_{[d]}$ be all subgradient magnitudes in increasing order. Given k^* in (2), we define the *TopK adaptation*, which selects the largest subgradients, as follows:

$$\lambda^{(t+1)} = \begin{cases} |p^{(t+1)}|_{[d-\mathsf{k}^*]} & \text{if } t \bmod \Delta t = 0, \\ \lambda^{(t)} & \text{else.} \end{cases} \tag{8}$$

The threshold $\lambda^{(t+1)}$ equals the $(d-\mathsf{k}^*)$-th smallest subgradient magnitude, and the k^* largest magnitudes remain strictly above it, provided $|p^{(t+1)}|_{[d-\mathsf{k}^*+1]} > |p^{(t+1)}|_{[d-\mathsf{k}^*]}$ holds. With such $\lambda^{(t+1)}$, the proximal step in (6) chooses the k^* coordinates whose subgradients carry the largest magnitudes, and sets the remaining $d-\mathsf{k}^*$ coordinates to 0. The threshold therefore exactly enforces the sparsity constraint on the subsequent iterate.

Relationship to DST. Given the mask $m \in \{0,1\}^d$ with $|m|_0 = \mathsf{k}^*$, DST methods apply a train-prune-grow cycle in which they train the masked weights $m \odot \theta$ at time t as follows,

$$\theta^{(t+1)} = m^{(t)} \odot \big(\theta^{(t)} - \tau \nabla_\theta \mathcal{L}(m^{(t)} \odot \theta^{(t)})\big), \tag{9}$$

where $\odot$ is the Hadamard product and m is updated every $\Delta t_{\text{mask}} \in \mathbb{N}$ iterations. The prune stage drops a fraction of active weights with the smallest weight magnitudes $|\theta_i|$. The grow stage adds as many weights as were dropped, either randomly or based on other criteria, e.g., the largest gradients per batch (Evci et al., 2020; Liu et al., 2021a) or momentum (Dettmers & Zettlemoyer, 2019).

Our proposed TopK adaptation method using LinBreg replaces the masked SGD step in (9) with the MD step in (6). To allow a comparison with DST, we define the active coordinates through the mask $m_i^{(t)} := \mathbf{1}\{|p_i^{(t)}| > \lambda^{(t)}\}$. Proposition 2 maps every stage of the DST cycle onto (6), using two auxiliary results in Lemma 3. Note that the equivalence with the DST cycle as presented holds solely for $\Delta t_{\text{mask}} = 1$, since, without additional control, adaptive LinBreg masks may change after each proximal step.

Lemma 3. *Assuming LinBreg iterates with $m_i^{(t)} = \mathbf{1}\{|p_i^{(t)}| > \lambda^{(t)}\}$, then*

(a) $\theta_i^{(t+1)} = m_i^{(t+1)}\big(p_i^{(t+1)} - \lambda^{(t+1)} \operatorname{sign}(p_i^{(t+1)})\big)$,

(b) $\forall\, i, j$ *with* $m_i^{(t)} = m_j^{(t)} = 1$: $\quad |\theta_i^{(t)}| > |\theta_j^{(t)}| \iff |p_i^{(t)}| > |p_j^{(t)}|$.

Part (a) formulates a LinBreg proximal step using binary masks. Part (b) states that the active weights and their associated subgradients induce the same ranking by magnitude.

Proposition 2. *Let $\lambda^{(t)}$ follow the adaptation rule* (8)*, then LinBreg with TopK adaptation performs the train-prune-grow cycle of DST as follows.*

(i) Every $\theta_i^{(t+1)} \neq 0$ accumulates the gradient history up to iteration t

$$|\theta_i^{(t+1)}| = |p_i^{(0)} - \tau \sum_{u=0}^{t} \nabla_{\theta_i} \mathcal{L}(\theta^{(u)})| - \lambda^{(t+1)} \qquad \textit{for } |p_i^{(t+1)}| > \lambda^{(t+1)}.$$

(ii) If $m_i^{(t)} \operatorname{sign}(p_i^{(t)}) = m_i^{(t+1)} \operatorname{sign}(p_i^{(t+1)})$*, then the update step is defined as*

$$\theta_i^{(t+1)} = m_i^{(t)} \big(\theta_i^{(t)} - \tau \nabla_{\theta_i} \mathcal{L}(\theta^{(t)}) - \operatorname{sign}(\theta_i^{(t)}) (\lambda^{(t+1)} - \lambda^{(t)}) \big) \tag{10}$$

(iii) If $m_i^{(t)} \operatorname{sign}(p_i^{(t)}) \neq m_i^{(t+1)} \operatorname{sign}(p_i^{(t+1)})$*, then*

$$0 < \max(\big|\theta_i^{(t)}\big|, \big|\theta_i^{(t+1)}\big|) \leq \big|\lambda^{(t+1)} - \lambda^{(t)}\big| + \tau \big|\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})\big| \tag{11}$$

(iv) If $|p^{(t+1)}|_{[d-\mathsf{k}^*+1]} > |p^{(t+1)}|_{[d-\mathsf{k}^*]}$*, then* $\lambda^{(t+1)} = |p^{(t+1)}|_{[d-\mathsf{k}^*]}$ *yields* $\mathsf{s}(\theta^{(t+1)}) = \mathsf{s}^*$.

Part (i) characterizes our single growth/pruning criterion, which activates a network weight $\theta_i^{(t+1)}$ when $|p_i^{(t+1)}| > \lambda^{(t+1)}$ is satisfied and prunes it otherwise. Since $p^{(t+1)}$ accumulates all gradients up to iteration t, growth is guided by the *full gradient history* rather than by a single batch gradient. Furthermore, Lemma 3 (b) establishes that selecting weights with largest $|\theta^{(t+1)}|$ (i.e., the DST criterion) is equivalent to a selection based on $|p^{(t+1)}|$, which is our proposed criterion.

Part (ii) defines the update rule for coordinates that are either inactive at two consecutive iterations, or active at both without a sign change in p_i. By construction, (8) keeps $\lambda^{(t+1)}{=}\lambda^{(t)}$ for $\Delta t{-}1$ out of every Δt iterations; on these the update coincides with the masked SGD step (9). At an adaptation step, active parameters are shifted by the adaptation difference (cf. Figure 8). Part (iii) covers the remaining coordinates not covered by (ii), and gives them upper and lower bounds. Specifically, newly activated coordinates (or ones that flip sign) are *strictly non-zero* and their initial values are bounded from above by the bound in (11).

Finally, (iv) provides a guarantee for satisfying $\mathsf{s}(\theta^{(t+1)}) = \mathsf{s}^*$ at every adaptation step, provided the k^*-th and $(\mathsf{k}^*{+}1)$-th largest magnitudes of $p^{(t+1)}$ are different. Excluding this tie leaves exactly the k^* largest magnitudes strictly above $\lambda^{(t+1)}$. Proofs for Lemma 3 and Proposition 2 are presented in Appendices A.3 and A.4, respectively.

2.4.2 SIGNED GAP ADAPTATION

We propose an alternative heuristic scheme that does not require finding $|p^{(t)}|_{[d-\mathsf{k}^*]}$ to perform the adaptation step. The idea behind it is simple: increase $\lambda^{(t+1)}$ if $\mathsf{s}(\theta^{(t)}) < \mathsf{s}^*$, and decrease it if $\mathsf{s}(\theta^{(t)}) > \mathsf{s}^*$, with an update size proportional to their difference. Concretely, let $\epsilon^{(t)} := \mathsf{s}^* - \mathsf{s}(\theta^{(t)})$ be the current sparsity gap, then we propose the following update based on $\epsilon^{(t)}$

$$\lambda^{(t+1)} = \begin{cases} \lambda^{(t)} \big(1 + \alpha^{C^{(t)}} |\epsilon^{(t)}|\big)^{\operatorname{sign}(\epsilon^{(t)})} & \text{if } t \bmod \Delta t = 0, \\ \lambda^{(t)} & \text{else,} \end{cases} \tag{12}$$

where $C^{(t)} \geq 0$ is the number of times $\mathsf{s}(\theta^{(t)})$ crosses s^* and α is close to but smaller than 1 ($1 > \alpha \gg 0$). When sparsity converges and the target s^* is repeatedly crossed, $\alpha^{C^{(t)}}$ exponentially decays to 0, which stops the adaptation and avoids high-frequency oscillations in the vicinity of the target when $\mathsf{s}(\theta^{(t)}) \approx \mathsf{s}^*$. We present a brief analysis of the algorithm's properties in Appendix C. Both our adaptation schemes are summarized in Algorithm 1.

2.4.3 TARGET SPARSITY SCHEDULING

We extend our sparsity-constrained MD by combining our proposed method with a moving sparsity target such that multiple sparsity constraints can be satisfied at different training stages. This implies adapting $\lambda^{(t)}$ to satisfy user-controlled sparsity schedules over defined number of iterations or epochs. We define our sparsity constraints based on the schedule presented in (Zhu & Gupta, 2017) as follows

$$\mathsf{s}^{(t)} = \mathsf{s}^* + \Big(\mathsf{s}(\theta^{(0)}) - \mathsf{s}^*\Big) \left(1 - \frac{t}{n\Delta t}\right)^3 \quad \text{for } t \in \{0, \Delta t, \ldots, n\Delta t\}, \tag{13}$$

where $\mathsf{s}^{(t)}$ is intermediate target sparsity at iteration t and n is the length of the schedule.

Algorithm 1: Adaptive LinBreg iterations with TopK and SGap adaptations

Input: Target sparsity s^*, adaptation frequency Δt, learning rate τ, adaptation factor α (only for method 2)
Initialize: $\lambda^{(0)}$, $p^{(0)}{=}\theta^{(0)}{+}\lambda^{(0)}\,\mathrm{sign}(\theta^{(0)})$. For method 2: $\epsilon \leftarrow \mathsf{s}^*{-}\mathsf{s}\big(\theta^{(0)}\big)$, $C{\leftarrow}0$
for $t = 0, 1, 2, \ldots$ **do**
  $p \leftarrow p - \tau\nabla\mathcal{L}(\theta)$ // Dual step
  if $t \bmod \Delta t = 0$ **then**
    Method 1: TopK subgradients
    $\mathsf{k}^* = d(1 - \mathsf{s}^*)$
    $\lambda \leftarrow |p|_{[d-\mathsf{k}^*]}$ // $(\mathsf{k}^*{+}1)$-th largest $|p_i|$

    Method 2: Signed gap
    $\epsilon_{\text{prev}} \leftarrow \epsilon$
    $\epsilon \leftarrow \mathsf{s}^* - \mathsf{s}(\theta)$
    if $\epsilon \cdot \epsilon_{prev} < 0$ **then** $C \leftarrow C + 1$
    $\lambda \leftarrow \lambda(1 + \alpha^C|\epsilon|)^{\mathrm{sign}(\epsilon)}$
  $\theta \leftarrow \mathrm{sign}(p)\max(|p| - \lambda, 0)$ // Proximal step

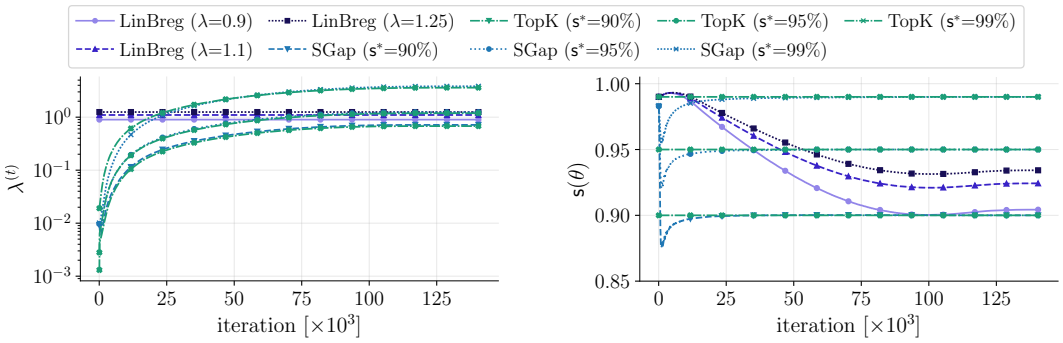


Figure 1: Evolution of sparsity and $\lambda^{(t)}$ measured on ResNet-50 and TinyImageNet for different s^*.

3 EXPERIMENTAL SETUP

We report the experimental results of our proposed methods on three image classification benchmarks: CIFAR-100 (Krizhevsky, 2009), TinyImageNet (Le & Yang, 2015), and ImageNet-1K (Russakovsky et al., 2015). We used WideResNet-28-10 (WRN-28-10) (Zagoruyko & Komodakis, 2017) and ResNet-50 (He et al., 2016) network architectures. On CIFAR-100 and TinyImageNet we trained for 200 epochs with a weight decay of $5{\times}10^{-4}$ and a batch size of 128. On ImageNet-1K, we trained for 100 epochs with a weight decay of $1{\times}10^{-4}$ and a batch size of 256. The learning rate was warmed up from 0.01 to 0.1 over 10% of the epochs, and then decayed by a cosine schedule. The dataset details and the remaining hyperparameters are reported in Appendix D.

We refer to LinBreg with TopK adaptation simply as TopK, and to the signed gap adapted LinBreg as SGap. We use $\lambda^{(0)}{=}0.01$ and $\alpha{=}0.95$ (in case of SGap) for all models, datasets, and target sparsity levels, setting $\Delta t{=}50$ for all experiments except when stated otherwise. Both the adaptive and the non-adaptive methods were initialized at $\mathsf{s}(\theta^{(0)}){=}99\%$. For the sparsity-scheduled variants, we used the schedule in (13) and set $\mathsf{s}(\theta^{(0)}){=}50\%$ to match GraNet (Liu et al., 2021a). The schedule length n was set to half of the training duration, i.e., the models reach the final sparsity target s^* mid-training.

4 RESULTS

Reaching target sparsity s^*: Figure 1 shows the evolution of sparsity and $\lambda^{(t)}$ for ResNet-50 on TinyImageNet at different values of s^*. The curves illustrate that TopK adaptation fulfills the target sparsity at the start of training and maintains it throughout the whole process. For SGap, sparsity starts at 99%, drops while λ is still small at its initial value, then quickly adapts towards s^*. We further supply the curves for the additional cases of sparsity scheduling and on AdaBreg (with ResNet-18) in Figures 3 and 5 in Appendix E. Our results show that every studied adaptive setting satisfied the constraint (7), supporting our sparsity control claim.

Table 1: Top-1 test accuracy in [%]. Gray and bold values indicate the best accuracy, gray-only indicates the next best value per sparsity rate. 'Ramp' refers to using sparsity scheduling. The sparsity at the best checkpoint for LinBreg is shown as $\mathsf{s}(\theta^*)$.

Method	CIFAR-100			TinyImageNet		
	90%	95%	99%	90%	95%	99%
ResNet-50 (Dense)		$80.0_{\pm 0.2}$			$69.6_{\pm 0.2}$	
RigL	$78.5_{\pm 0.3}$	$77.2_{\pm 0.4}$	$72.7_{\pm 0.3}$	$66.8_{\pm 0.3}$	$64.7_{\pm 0.1}$	$61.3_{\pm 0.2}$
GraNet	$78.9_{\pm 0.4}$	$78.0_{\pm 0.4}$	$75.1_{\pm 0.2}$	$67.7_{\pm 0.4}$	$66.3_{\pm 0.2}$	$64.0_{\pm 0.4}$
LinBreg ($\lambda = 0.90/1.10/1.25$)	$75.1_{\pm 0.4}$	$74.5_{\pm 0.1}$	$73.7_{\pm 0.3}$	$64.6_{\pm 0.4}$	$63.8_{\pm 0.1}$	$63.4_{\pm 0.5}$
$\mathsf{s}(\theta^*)$	$95.3_{\pm 0.0}$	$96.3_{\pm 0.0}$	$96.9_{\pm 0.0}$	$90.4_{\pm 0.1}$	$92.4_{\pm 0.0}$	$93.4_{\pm 0.0}$
TopK (ours)	$78.0_{\pm 0.4}$	$77.6_{\pm 0.4}$	$75.1_{\pm 0.2}$	$66.9_{\pm 0.2}$	$65.5_{\pm 0.4}$	$62.5_{\pm 0.3}$
SGap (ours)	$78.3_{\pm 0.3}$	$77.5_{\pm 0.2}$	$75.1_{\pm 0.6}$	$67.2_{\pm 0.3}$	$66.0_{\pm 0.6}$	$63.6_{\pm 0.3}$
TopK + Ramp (ours)	$78.9_{\pm 0.0}$	$\mathbf{78.7_{\pm 0.2}}$	$\mathbf{76.9_{\pm 0.2}}$	$\mathbf{68.2_{\pm 0.6}}$	$\mathbf{67.1_{\pm 0.2}}$	$65.7_{\pm 0.5}$
SGap + Ramp (ours)	$\mathbf{79.2_{\pm 0.4}}$	$\mathbf{78.7_{\pm 0.1}}$	$\mathbf{76.9_{\pm 0.2}}$	$67.9_{\pm 0.1}$	$\mathbf{67.1_{\pm 0.4}}$	$\mathbf{66.4_{\pm 0.3}}$
WRN-28-10 (Dense)		$79.9_{\pm 0.2}$			$68.3_{\pm 0.1}$	
RigL	$79.6_{\pm 0.2}$	$78.8_{\pm 0.3}$	$75.2_{\pm 0.2}$	$66.4_{\pm 0.1}$	$65.0_{\pm 0.0}$	$61.8_{\pm 0.2}$
GraNet	$\mathbf{79.9_{\pm 0.3}}$	$\mathbf{79.3_{\pm 0.1}}$	$77.0_{\pm 0.1}$	$66.4_{\pm 0.7}$	$65.7_{\pm 0.2}$	$64.4_{\pm 0.2}$
LinBreg ($\lambda = 0.90/1.10/1.25$)	$76.1_{\pm 0.1}$	$75.9_{\pm 0.4}$	$75.5_{\pm 0.3}$	$62.8_{\pm 0.1}$	$62.3_{\pm 0.2}$	$61.8_{\pm 0.3}$
$\mathsf{s}(\theta^*)$	$95.8_{\pm 0.0}$	$96.7_{\pm 0.0}$	$97.2_{\pm 0.0}$	$91.7_{\pm 0.0}$	$93.3_{\pm 0.0}$	$94.2_{\pm 0.0}$
TopK (ours)	$79.2_{\pm 0.3}$	$79.1_{\pm 0.2}$	$76.0_{\pm 0.2}$	$66.1_{\pm 0.2}$	$64.9_{\pm 0.1}$	$62.5_{\pm 0.4}$
SGap (ours)	$79.6_{\pm 0.1}$	$79.0_{\pm 0.0}$	$77.0_{\pm 0.3}$	$66.2_{\pm 0.3}$	$64.9_{\pm 0.1}$	$63.0_{\pm 0.3}$
TopK + Ramp (ours)	$\mathbf{79.9_{\pm 0.4}}$	$79.2_{\pm 0.2}$	$\mathbf{77.9_{\pm 0.2}}$	$66.5_{\pm 0.4}$	$65.8_{\pm 0.4}$	$65.3_{\pm 0.2}$
SGap + Ramp (ours)	$79.8_{\pm 0.4}$	$\mathbf{79.3_{\pm 0.1}}$	$77.7_{\pm 0.4}$	$\mathbf{66.8_{\pm 0.2}}$	$\mathbf{66.0_{\pm 0.3}}$	$\mathbf{65.6_{\pm 0.4}}$

Accuracy: Table 1 reports test accuracy on CIFAR-100 and TinyImageNet and Table 2 reports it on ImageNet-1K. The fixed λ runs reach sparsity rates of between 90–97% depending on the model and dataset. This means a one-to-one comparison between our TopK and SGap variants and LinBreg is not always possible. However, the tables show that our adaptive optimizers outperform every LinBreg run at comparable or higher sparsity, with particularly large improvement when sparsity scheduling is utilized. For example, at $\mathsf{s}(\theta)$=99%, Table 1 reports ResNet-50 with sparsity-scheduled TopK and SGap achieve accuracies of 65.7% and 66.4% on TinyImageNet, respectively, outperforming LinBreg at only 90.4% sparsity (accuracy = 64.6%). TopK and SGap without sparsity scheduling still dominate LinBreg on TinyImageNet but at s=95% instead of s=99%. Table 2 similarly shows that on ImageNet-1K, our optimizers improved accuracy even at higher sparsity compared to LinBreg, e.g., 72.5% at 95% sparsity versus 70.5% at only 91%. These results imply that λ adaptation not only facilitates sparsity control, but may also lead to significant performance improvement. Our methods with sparsity scheduling also demonstrate superior performance compared to the DST baselines on the evaluated benchmarks. Finally, we observe that SGap and TopK show on par performance across benchmarks. This means that finding the exact $\lambda^{(t)}$ that satisfies s^* is not strictly required for adaptation; it only suffices to steer it towards the target, e.g., as proposed with SGap.

Topology exploration: Sparsity alone does not fully describe the pattern of weight activity in a network. A sparse network can maintain $|m^{(t)}|_0$=k^* while either constantly replacing $m^{(t)}$, or keeping it fixed. While DST frameworks explicitly control $m^{(t)}$, the mask control is only implicit in our MD setup as the masks are defined by proximal steps. To understand the training dynamics from the mask, we define $\rho^{(e)}$ at epoch e as an estimate of the mask replacement rate,

$$\rho^{(e)} = \frac{\left|m^{(e)} - m^{(e-1)}\right|_0}{\left|m^{(e)}\right|_0 + \left|m^{(e-1)}\right|_0}. \tag{14}$$

The numerator counts the parameters that changed from $e-1$ to e, and the denominator is the number of active coordinates at $e-1$ and e (= $2\mathsf{k}^*$ for constant sparsity). Hence, $\rho^{(e)}$ estimates the fraction of the active weights replaced in an epoch and measured at the epoch boundary. Notice that $\rho^{(e)}$ does not account for the activity between iterations within a single epoch. Here we use $\rho^{(e)}$ and its cumulative sum as a proxy for topological exploration, related to the ITOP rate in Liu et al. (2021b). As they show (and we argue below), exploration alone does not imply improved performance, e.g., when newly activated weights are pruned before they are sufficiently trained.

Figure 2 reports $\rho^{(e)}$ over the training duration. Across all methods, three main observations can be drawn. First, adaptive λ yields higher values of $\rho^{(e)}$ than a fixed λ over nearly all epochs. Second,

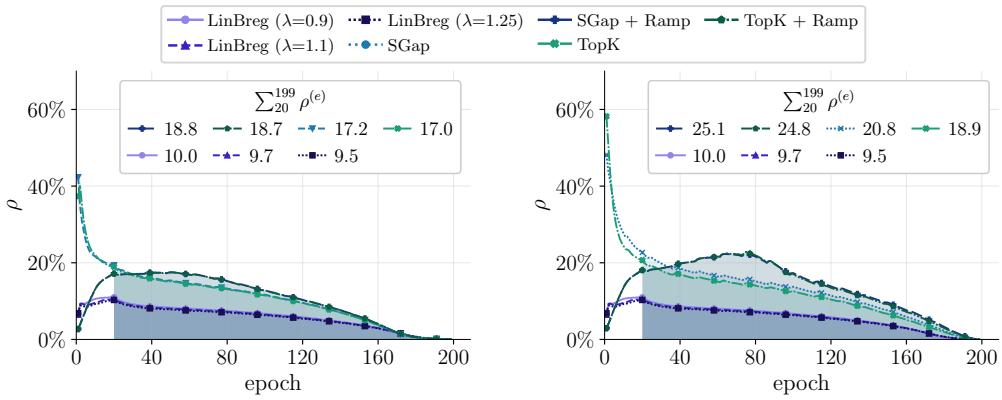


Figure 2: ρ for ResNet-50 on TinyImageNet at s^*=90% (left) and s^*=99% (right).

both our TopK and SGap report similar $\rho^{(e)}$ values, both with and without sparsity scheduling. Third, compared to fixed target, gradual sparsity adaptation leads to smaller $\rho^{(e)}$ at the start of training, then quickly increases it for nearly all remaining epochs. The reason why $\rho^{(e)}$ is relatively high in the first epochs for a fixed s^* is the initialization, which sets $p_i^{(0)}=0$ for the 99% of coordinates with $\theta_i^{(0)}=0$. In early iterations, the magnitudes $|p_i^{(t)}|$ of these coordinates are thus determined only by their accumulated gradients, which are small and densely clustered, so small gradient fluctuations easily sway many of them across the threshold $\lambda^{(t)}$, especially since $|\nabla\mathcal{L}|$ is typically largest at the start of training. Due to the described dynamics, the high rate of mask updates after initialization may not be a reliable signal for quantifying exploration. Past this initial phase (the first 10% of epochs), however, the cumulative sum $\sum\rho^{(e)}$ largely follows the ranking of the methods by test accuracy in Table 1. Interestingly, the gap in $\sum\rho^{(e)}$ between our scheduled and unscheduled variants also widens with larger s^*, as does their accuracy gap. This is consistent with exploration contributing to the accuracy gains for both λ adaptation and sparsity scheduling.

Table 2: Top-1 accuracy [%] and normalized FLOPs of ResNet-50 on ImageNet-1K.

	90%			95%		
Method	Acc.	FLOPs/Train	FLOPs/Test	Acc.	FLOPs/Train	FLOPs/Test
Dense	$77.1_{\pm0.1}$	1×(3.1e18)	1×(8.2e9)	$77.1_{\pm0.1}$	1×(3.1e18)	1×(8.2e9)
GMP	$70.9_{\pm0.1}$	0.25	0.13	$68.5_{\pm0.1}$	0.20	0.07
SET (ERK)	$73.5_{\pm0.1}$	0.24	0.24	$70.1_{\pm0.2}$	0.12	0.12
RigL (ERK)	$73.6_{\pm0.1}$	0.24	0.24	$70.3_{\pm0.1}$	0.12	0.12
GraNet	$74.2_{\pm0.1}$	0.30	0.22	$71.7_{\pm0.1}$	0.21	0.12
LinBreg (λ=2/4)	$70.5_{\pm0.1}$	0.40	0.13	$65.8_{\pm0.1}$	0.37	0.07
$\mathsf{s}(\theta^*)$		$91.0_{\pm0.0}$			$95.6_{\pm0.0}$	
TopK	$74.2_{\pm0.1}$	0.43	0.14	$72.1_{\pm0.0}$	0.40	0.08
SGap	$74.0_{\pm0.1}$	0.42	0.13	$71.9_{\pm0.1}$	0.39	0.08
TopK + Ramp	$\mathbf{74.4}_{\pm\mathbf{0.0}}$	0.47	0.15	$\mathbf{72.5}_{\pm\mathbf{0.1}}$	0.43	0.09
SGap + Ramp	$\mathbf{74.4}_{\pm\mathbf{0.1}}$	0.47	0.15	$72.4_{\pm0.1}$	0.44	0.09

5 CONCLUSIONS

This work promotes adaptive regularization as a means of controlling sparsity in the mirror descent optimization framework. We presented two adaptation schemes, namely TopK and SGap. We analyzed both and showed that TopK adaptation admits a DST-like train-prune-grow interpretation. Furthermore, we extended the sparsity control with sparsity scheduling to gradually increase sparsity towards the target. We validated our proposed algorithms on the benchmarks CIFAR-100, TinyImageNet, and ImageNet-1K with ResNet-50 and WideResNet-28-10. We showed that both algorithms reliably steer the models towards a user-specified sparsity, as well as to the intermediate targets set by sparsity schedules. In addition to sparsity control, our methods achieved higher top-1 accuracy compared to the fixed λ of the original scheme, especially when combined with a sparsity schedule. The scheduled variants also match or outperform the baseline DST methods on the evaluated benchmarks.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods, to provide feedback on our experimental methodology, and to assist in the writing of proofs. Specifically, they reviewed our mathematical statements and proofs, which led to corrections of some statements and proof steps, and shortened part of the proof of Proposition 2. They also contributed to the interpretation of some results. We did not use generative AI tools to generate synthetic data, to develop the theoretical framework, to formulate the mathematical claims, to provide critical ingredients for proving mathematical claims, to propose or refine hypotheses, for translation, or to clean and reformat datasets. Qualitative and thematic data analysis is not applicable to this work. Additionally, we used generative AI tools to search for, identify, and summarize related literature and to edit the manuscript for improved readability. However, the first draft was written by the authors. All AI-assisted work was reviewed by at least one author, and AI-generated code was verified with a range of tests. We take responsibility for the final content of this work, including all text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. We use only the publicly available benchmarks CIFAR-100, TinyImageNet, and ImageNet-1K. The proposed contribution is methodological and entails no foreseeable risk of misuse beyond that inherent to standard image classification.

REPRODUCIBILITY STATEMENT

Theoretical claims are stated with their assumptions and proved in full in the Appendix. The architectures, training recipe, and hyperparameters are specified in Section 3 and Appendix D, which additionally list the GPUs used in the experiments. The three datasets we used are publicly available, and the splits and training length used for each are explicitly stated. All results are reported as mean $\pm$ standard deviation over three random seeds. Our code is included in the anonymous supplementary material.

REFERENCES

Hédy Attouch, Jérôme Bolte, Patrick Redont, and Antoine Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-łojasiewicz inequality. *Mathematics of operations research*, 35(2):438–457, 2010.

Hedy Attouch, Jérôme Bolte, and Benar Fux Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods. *Mathematical programming*, 137(1):91–129, 2013.

Navid Azizan, Sahin Lale, and Babak Hassibi. Stochastic mirror descent on overparameterized nonlinear models. *IEEE Trans. Neural Nets. and Learning Systems*, 33(12):7717–7727, 2022.

Heinz H. Bauschke and Patrick L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer International Publishing, 2017. ISBN 9783319483115. doi: 10.1007/978-3-319-48311-5. URL http://dx.doi.org/10.1007/978-3-319-48311-5.

Alessandro Benfenati and Valeria Ruggiero. Inexact Bregman iteration with an application to poisson data reconstruction. *Inverse Problems*, 29(6):065016, 2013.

Martin Benning, Marta M. Betcke, Matthias J. Ehrhardt, and Carola-Bibiane Schönlieb. Choose your path wisely: Gradient descent in a bregman distance framework. *SIAM Journal on Imaging Sciences*, 14(2):814–843, 2021. doi: 10.1137/20M1357500.

Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1):459–494, 2014.

Leon Bungert, Tim Roith, Daniel Tenbrinck, and Martin Burger. A Bregman learning framework for sparse neural networks. *JMLR*, 23(192):1–43, 2022. URL http://jmlr.org/papers/v23/21-0545.html.

Jian-Feng Cai, Stanley Osher, and Zuowei Shen. Linearized Bregman iterations for compressed sensing. *Mathematics of computation*, 78(267):1515–1536, 2009.

Tim Dettmers and Luke Zettlemoyer. Sparse networks from scratch: Faster training without losing performance, 2019. URL https://arxiv.org/abs/1907.04840.

P. Dhar. The carbon impact of artificial intelligence. *Nat. Mach. Intell.*, 2:423–425, 2020.

Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the lottery: Making all tickets winners. In *Proc. ICML*, pp. 2943–2952, 2020.

Pierre Frankel, Guillaume Garrigos, and Juan Peypouquet. Splitting methods with variable metric for kurdyka–łojasiewicz functions and general convergence rates. *Journal of Optimization Theory and Applications*, 165(3):874–900, 2015.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *Proc. ICLR*, 2019. URL https://openreview.net/forum?id=rJl-b3RcF7.

Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks, 2019. URL https://arxiv.org/abs/1902.09574.

Jose Gallego-Posada, Juan Ramirez, Akram Erraqabi, Yoshua Bengio, and Simon Lacoste-Julien. Controlled sparsity via constrained optimization or: How i learned to stop tuning penalties and love constraints. In *Proc. NeurIPS*, 2022. URL https://openreview.net/forum?id=XUvSYc6TqDF.

Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Characterizing implicit bias in terms of optimization geometry. In *Proc. ICML*, pp. 1832–1841. PMLR, 2018.

S. Han, J. Pool, J. Tran, and W. Dally. Learning both weights and connections for efficient neural networks. In *Proc. NeurIPS*, volume 28, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, June 2016.

T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden, and A. Peste. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *JMLR*, 22(241):1–124, 2021.

Chendi Huang, Xinwei Sun, Jiechao Xiong, and Yuan Yao. Split LBI: An iterative regularization path with structural sparsity. In *Proc. NeurIPS*, pp. 3369–3377, 2016.

Siddhant M Jayakumar, Razvan Pascanu, Jack W Rae, Simon Osindero, and Erich Elsen. Top-KAST: Top-K always sparse training. In *Proc. NeurIPS*, volume 33, pp. 20744–20754, 2020.

Jie Ji, Gen Li, Lu Yin, Minghai Qin, Geng Yuan, Linke Guo, Shiwei Liu, and Xiaolong Ma. Advancing dynamic sparse training by exploring optimization opportunities. In *Proc. ICML*, volume 235, pp. 21606–21619, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/ji24a.html.

Junghoe Kim, Vince D. Calhoun, Eunsoo Shim, and Jong-Hwan Lee. Deep neural network with weight sparsity control and pre-training extracts hierarchical features and enhances classification performance: Evidence from whole-brain resting-state functional connectivity patterns of schizophrenia. *NeuroImage*, 124:127–146, 2016.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. ICLR*, 2015.

Alex Krizhevsky. Learning multiple layers of features from tiny images, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Aditya Kusupati, Vivek Ramanujan, Raghav Somani, Mitchell Wortsman, Prateek Jain, Sham Kakade, and Ali Farhadi. Soft threshold weight reparameterization for learnable sparsity. In *Proc. ICML*, July 2020.

Ya Le and Xuan Yang. Tiny ImageNet visual recognition challenge, 2015. URL http://vision.stanford.edu/teaching/cs231n/reports/2015/pdfs/yle_project.pdf.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip Torr. SNIP: Single-shot network pruning based on connection sensitivity. In *Proc. ICLR*, 2019. URL https://openreview.net/forum?id=B1VZqjAcYX.

Gen Li, Lu Yin, Jie Ji, Wei Niu, Minghai Qin, Bin Ren, Linke Guo, Shiwei Liu, and Xiaolong Ma. NeurRev: Train better sparse neural network practically via neuron revitalization. In *Proc. ICLR*, 2024. URL https://openreview.net/forum?id=60lNoatp7u.

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. In *Proc. NeurIPS*, 2021a.

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 *Proc. ICML*, volume 139, pp. 6989–7000. PMLR, July 2021b. URL https://proceedings.mlr.press/v139/liu21y.html.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proc. ICLR*, 2017. URL https://api.semanticscholar.org/CorpusID:53592270.

Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l_0 regularization. In *Proc. ICLR*, 2018.

Alexandra Sasha Luccioni, Sylvain Viguier, and Anne-Laure Ligozat. Estimating the carbon footprint of BLOOM, a 176b parameter language model. *JMLR*, 24(253):1–15, 2023.

Yannick Lunk, Sebastian J. Scott, and Leon Bungert. Sparse training of neural networks based on multilevel mirror descent, 2026. URL https://arxiv.org/abs/2602.03535.

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 Comm.*, 9(1):2383, 2018.

A. S. Nemirovsky and D. B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. John Wiley & Sons, 1983.

Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical programming*, 120(1):221–259, 2009.

Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.

R Tyrrell Rockafellar and Roger JB Wets. *Variational Analysis*. Springer, 1998.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *IJCV*, 115:211–252, 2015. URL https://doi.org/10.1007/s11263-015-0816-y.

Jaime Sevilla, Lennart Heim, Anson Ho, Tamay Besiroglu, Marius Hobbhahn, and Pablo Villalobos. Compute trends across three eras of machine learning. In *Proc. IJCNN*, pp. 1–8, 2022.

Lixin Shen, Rui Wang, Yuesheng Xu, and Mingsong Yan. Sparse deep learning models with the ℓ_1 regularization. *arXiv preprint arXiv:2408.02801*, 2024.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. In *Proc. ACL*, pp. 3645–3650, 2019.

Yi-Lin Sung, Varun Nair, and Colin Raffel. Training neural networks with fixed sparse masks. In *Proc. NeurIPS*, volume 34, pp. 24193–24205, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/cb2653f548f8709598e8b5156738cc51-Paper.pdf.

Hidenori Tanaka, Daniel Kunin, Daniel Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. In *Proc. NeurIPS*, volume 33, pp. 6377–6389, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/46a4378f835dc8040c8057beb6a2da52-Paper.pdf.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Jour. Roy. Stat. Soc. Series B*, 58(1): 267–288, 1996. URL http://www.jstor.org/stable/2346178.

Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. In *Proc. ICLR*, 2020. URL https://openreview.net/forum?id=SkgsACVKPH.

Xiaoyu Wang and Martin Benning. Lifted Bregman training of neural networks. *JMLR*, 24(232): 1–51, 2023.

Wotao Yin, Stanley Osher, Donald Goldfarb, and Jerome Darbon. Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1 (1):143–168, 2008.

Xin Yuan, Pedro Henrique Pamplona Savarese, and Michael Maire. Growing efficient deep networks by structured continuous sparsification. In *Proc. ICLR*, 2021. URL https://openreview.net/forum?id=wb3wxCObbRT.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks, 2017. URL https://arxiv.org/abs/1605.07146.

Xiaoqun Zhang, Martin Burger, Xavier Bresson, and Stanley Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM Journal on Imaging Sciences*, 3(3): 253–276, 2010.

Michael Zhu and Suyog Gupta. To prune, or not to prune: exploring the efficacy of pruning for model compression, 2017. URL https://arxiv.org/abs/1710.01878.

APPENDIX

A THEORY AND PROOFS

A.1 LOSS DECAY AND CONVERGENCE UP TO SUBSEQUENCES

We start by providing a proof of lemma 1.

Proof. The L-smoothness of $\mathcal{L}$ implies

$$\mathcal{L}(\theta^{(t+1)}) - \mathcal{L}(\theta^{(t)}) - \frac{L}{2}|\theta^{(t+1)} - \theta^{(t)}|_2^2 \leq \langle \nabla \mathcal{L}(\theta^{(t)}), \theta^{(t+1)} - \theta^{(t)} \rangle$$
$$= -\frac{1}{\tau}\langle p^{(t+1)} - p^{(t)}, \theta^{(t+1)} - \theta^{(t)} \rangle.$$

By definition of the scheme, we have that $p^{(t)} \in \partial \phi_{\lambda^{(t)}}(\theta^{(t)})$ for every t and thus there exist $g^{(t)} \in \partial |\cdot|_1(\theta^{(t)})$ such that

$$p^{(t)} = \theta^{(t)} + \lambda^{(t)} g^{(t)},$$

which yields

$$\langle p^{(t+1)} - p^{(t)}, \theta^{(t+1)} - \theta^{(t)} \rangle = |\theta^{(t+1)} - \theta^{(t)}|_2^2 + \langle \lambda^{(t+1)} g^{(t+1)} - \lambda^{(t)} g^{(t)}, \theta^{(t+1)} - \theta^{(t)} \rangle.$$

Furthermore, using that $g^{(t)} \in \partial |\cdot|_1(\theta^{(t)})$ we get

$$\langle g^{(t)}, \theta^{(t)} \rangle = |\theta^{(t)}|_1$$
$$\langle g^{(t)}, v \rangle \leq |g^{(t)}|_\infty |v|_1 \leq |v|_1, \qquad \forall v \in \mathbb{R}^d$$

where we used Hölder's inequality in the second line. With this we further estimate,

$$\begin{aligned}
&\langle \lambda^{(t+1)} g^{(t+1)} - \lambda^{(t)} g^{(t)}, \theta^{(t+1)} - \theta^{(t)} \rangle \\
&= \lambda^{(t+1)} |\theta^{(t+1)}|_1 + \lambda^{(t)} |\theta^{(t)}|_1 - \lambda^{(t+1)} \langle g^{(t+1)}, \theta^{(t)} \rangle - \lambda^{(t)} \langle g^{(t)}, \theta^{(t+1)} \rangle \\
&\geq \lambda^{(t+1)} |\theta^{(t+1)}|_1 + \lambda^{(t)} |\theta^{(t)}|_1 - \lambda^{(t+1)} |\theta^{(t)}|_1 - \lambda^{(t)} |\theta^{(t+1)}|_1 \\
&= (\lambda^{(t+1)} - \lambda^{(t)})(|\theta^{(t+1)}|_1 - |\theta^{(t)}|_1).
\end{aligned}$$

Finally, we obtain

$$\mathcal{L}(\theta^{(t+1)}) - \mathcal{L}(\theta^{(t)}) - \frac{L}{2}|\theta^{(t+1)} - \theta^{(t)}|_2^2 \leq$$
$$- \frac{1}{\tau}|\theta^{(t+1)} - \theta^{(t)}|_2^2 - \frac{\lambda^{(t+1)} - \lambda^{(t)}}{\tau}(|\theta^{(t+1)}|_1 - |\theta^{(t)}|_1).$$

□

With this we can now give a proof of lemma 2. In fact it will be convenient to slightly extend its assertion and we will prove the following lemma.

Lemma 4. *Let assumption 1 hold true, $\tau < \frac{2}{L}$ and let $(\lambda^{(t)})_{t\in\mathbb{N}}$ be such that $(\lambda^{(t+1)} - \lambda^{(t)})_{t\in\mathbb{N}} \in \ell^2(\mathbb{R})$, then the sequence $\theta^{(t)}$ is bounded, has a subsequence converging to some point $\theta^{(\infty)} \in \mathbb{R}^d$ and furthermore*

$$\sum_{t=0}^{\infty} |\theta^{(t+1)} - \theta^{(t)}|_2^2 < \infty.$$

Beyond that, the limit $\lim_{t\to\infty} \mathcal{L}(\theta^{(t)})$ exists in $\mathbb{R}$.

Proof. Starting from lemma 1 we first sum the inequality from $t = 0$ up to some $T \in \mathbb{N}$, which yields

$$\mathcal{L}(\theta^{(T+1)}) - \mathcal{L}(\theta^{(0)}) + \left(\frac{1}{\tau} - \frac{L}{2}\right) \sum_{t=0}^{T} |\theta^{(t+1)} - \theta^{(t)}|^2 \leq \sum_{t=0}^{T} \frac{\lambda^{(t+1)} - \lambda^{(t)}}{\tau}(|\theta^{(t)}|_1 - |\theta^{(t+1)}|_1).$$

For the term on the right hand side we introduce the short-hand notation $\Delta\lambda^{(t+1)} := \lambda^{(t+1)} - \lambda^{(t)}$ and further, we estimate

$$\left| |\theta^{(t+1)}|_1 - |\theta^{(t)}|_1 \right| \leq |\theta^{(t+1)} - \theta^{(t)}|_1 \leq \sqrt{d}\, |\theta^{(t+1)} - \theta^{(t)}|_2.$$

Together with an application of Young's inequality with some $\varepsilon > 0$ this yields

$$|\Delta\lambda^{(t+1)}| \left| |\theta^{(t)}|_1 - |\theta^{(t+1)}|_1 \right| \leq \frac{d|\theta^{(t+1)} - \theta^{(t)}|_2^2}{2\varepsilon} + \frac{(\Delta\lambda^{(t+1)})^2\varepsilon}{2},$$

Plugging this back into our first estimate, we obtain

$$\mathcal{L}(\theta^{(T+1)}) - \mathcal{L}(\theta^{(0)}) + \left(\frac{1}{\tau} - \frac{L}{2} - \frac{d}{2\varepsilon\tau}\right) \sum_{t=0}^{T} |\theta^{(t+1)} - \theta^{(t)}|^2 \leq \frac{\varepsilon}{2\tau} \sum_{t=0}^{T} (\Delta\lambda^{(t+1)})^2.$$

Next we choose $\varepsilon > 0$, such that $\nu := \left(\frac{1}{\tau} - \frac{L}{2} - \frac{d}{2\varepsilon\tau}\right) > 0$, which is possible due to the assumption that $\tau < \frac{2}{L}$. Using that $\left(\Delta\lambda^{(t)}\right)_{t\in\mathbb{N}} \in \ell^2(\mathbb{R})$ and that $\mathcal{L}$ is bounded from below, we can send $T \to \infty$ in the above bound to obtain

$$\limsup_{T\to\infty} \mathcal{L}(\theta^{(T)}) + \nu \sum_{t=1}^{\infty} |\theta^{(t+1)} - \theta^{(t)}|^2 \leq \frac{\varepsilon}{2\tau} \sum_{t=0}^{\infty} (\Delta\lambda^{(t)})^2 + \mathcal{L}(\theta^{(0)}) < \infty.$$

Using the fact that $\mathcal{L}$ has bounded sub-level sets, this implies that $(\theta^{(t)})_{t\in\mathbb{N}}$ is a bounded sequence in $\mathbb{R}^d$. Therefore, there exists a subsequence converging to some point $\theta^{(\infty)} \in \mathbb{R}^d$, i.e.,

$$\lim_{k\to\infty} \theta^{(t_k)} = \theta^{(\infty)}.$$

Beyond that, invoking lemma 1 once again, and disregarding the squared norm term, we get, that

$$\mathcal{L}(\theta^{(t+1)}) \leq \mathcal{L}(\theta^{(t)}) + \underbrace{\frac{d|\theta^{(t+1)} - \theta^{(t)}|_2^2}{2\varepsilon\tau} + \frac{(\Delta\lambda^{(t+1)})^2\varepsilon}{2\tau}}_{=:d^{(t)}}.$$

The sequence $f^{(t)} := \mathcal{L}(\theta^{(t)}) - \sum_{k=0}^{t-1} d^{(k)}$ then fulfills

$$f^{(t+1)} \leq \mathcal{L}(\theta^{(t)}) - \sum_{k=0}^{t-1} d^{(k)} = f^{(t)}.$$

Furthermore, since $\mathcal{L}$ is bounded from below and $d^{(t)}$ is summable, as show in the arguments above, we get that $f^{(t)}$ is non-increasing sequence that is bounded from below and thus is a convergent sequence. This yields that also $(\mathcal{L}(\theta^{(t)}))_{t\in\mathbb{N}}$ is a convergent sequence in $\mathbb{R}$. □

Note, that the above statement does not yet imply convergence of the full sequence. Our proof strategy below will require stronger assumptions on the sequence of parameters and the loss function itself. However, already now we want to see that if the sequence would converge, its limit is a stationary point of $\mathcal{L}$, which can be shown under even weaker conditions on the parameter sequence $(\lambda^{(t)})_{t\in\mathbb{N}}$.

Lemma 5. *Assume that $\nabla\mathcal{L}$ is continuous, that the sequence of parameters $(\lambda^{(t)})_{t\in\mathbb{N}}$ is bounded, that $(\lambda^{(t+1)} - \lambda^{(t)})_{t\in\mathbb{N}}$ is a null sequence and that the sequence $(\theta^{(t)})_{t\in\mathbb{N}}$ converges to some $\theta^{(\infty)}$, then this limit point is a stationary point of $\mathcal{L}$.*

Proof. We first plug in the iteration to see that

$$\tau\nabla\mathcal{L}(\theta^{(t)}) = p^{(t)} - p^{(t+1)} = \theta^{(t)} + \lambda^{(t)} g^{(t)} - \theta^{(t+1)} - \lambda^{(t+1)} g^{(t+1)}$$

with $g^{(t)} \in \partial|\cdot|_1(\theta^{(t)}), g^{(t+1)} \in \partial|\cdot|_1(\theta^{(t+1)})$. We now separate the cases where $\theta_i^{(\infty)}$ is zero or not for $i \in \{1, \ldots, d\}$.

First case $\theta_i^{(\infty)} \neq 0$: W.l.o.g. we assume that $\theta_i^{(\infty)} > 0$, then since the iteration converges, there exists some $T \in \mathbb{N}$, such that $\theta_i^{(t)} > 0$ for all $t \geq T$. This yields that $g_i^{(t)} = 1$ for all $t \geq T$ and thus

$$\tau|\nabla\mathcal{L}(\theta^{(t)})_i| \leq |\theta^{(t)} - \theta^{(t+1)}| + |\lambda^{(t)} - \lambda^{(t+1)}| \xrightarrow{t\to\infty} 0$$

where the last limit is due to the convergence of the iterates and the assumption on the sequence $(\lambda^{(t)})_{t\in\mathbb{N}}$. Since we assume $\nabla\mathcal{L}$ to be continuous this yields that $\nabla\mathcal{L}(\theta^{(\infty)})_i = 0$.

Second case $\theta_i^{(\infty)} = 0$: We assume the opposite, namely that $\nabla\mathcal{L}(\theta^{(\infty)})_i \neq 0$ and w.l.o.g. that $\nabla\mathcal{L}(\theta^{(\infty)})_i > 0$. Using the update formula this yields that there is some $C > 0$ and $T \in \mathbb{N}$ such that

$$\tau\nabla\mathcal{L}(\theta^{(t)})_i = p_i^{(t)} - p_i^{(t+1)} \geq C, \qquad \text{for all } t \geq T.$$

Unrolling this relationship until $t = T$ we obtain

$$p_i^{(t+1)} \leq p_i^{(T)} - C \cdot (t + 1 - T)$$

and thus $p_i^{(t)} \xrightarrow{t\to\infty} -\infty$. This however contradicts the fact that

$$\lim_{t\to\infty} \theta_i^{(t)} = \lim_{t\to\infty} \operatorname{prox}_{\lambda^{(t)}|\cdot|_1}(p^{(t)})_i = 0$$

since we assumed the sequence of regularization parameters to be bounded. Since we produced a contradiction, the assumption $\nabla\mathcal{L}(\theta^{(\infty)})_i > 0$ must have been impossible.

This finally yields that

$$\nabla\mathcal{L}(\theta^{(\infty)}) = 0.$$

□

A.2 CONVERGENCE OF THE ITERATES

We now turn to proving that the sequence of iterates does indeed converge, not only a subsequence. In order to do so, we follow the technique developed in Benning et al. (2021), which itself builds upon the theory from Attouch et al. (2013); Bolte et al. (2014). Following the concepts from Benning et al. (2021) we consider the following surrogate functional

$$\mathcal{F}(\theta, g) := \mathcal{L}(\theta) + \frac{\lambda^{(\infty)}}{\tau}|\theta|_1 + \chi_{\leq 1}(g) - \frac{\lambda^{(\infty)}}{\tau}\langle\theta, g\rangle$$

where $\chi_{\leq 1}$ is the indicator function on the set $B_1^\infty = \{g : |g|_\infty \leq 1\}$. Here, $\lambda^{(\infty)}$ is defined to be the limit of the parameter sequence, which we now assume to exist, i.e.,

$$\lim_{t\to\infty} \lambda^{(t)} = \lambda^{(\infty)} < \infty. \tag{15}$$

While Benning et al. (2021) also allows an adaptive regularization parameter, this is tied to the step size. In our case we want to make the parameter $\lambda^{(t)}$ independent of the step size, and therefore we cannot directly copy the results from Benning et al. (2021). Our analysis is adapted in a way, where an error term based on the changing regularization parameter appears. Assuming that this converges fast enough will allow us to still prove convergence. Concretely, in order to establish a finite-length property of the sequence $(\theta^{(t)})_{t\in\mathbb{N}}$ we will require that

$$|\lambda^{(t)} - \lambda^{(\infty)}| \leq C_\lambda t^q, \quad \text{with} \quad C_\lambda > 0 \text{ and } q < -\frac{3}{2}$$

and that $\mathcal{F}$ fulfills a Kurdyka–Łojasiewicz conditions. Not all the results in the following require such assumptions and we state them in a finer resolution below.

The first property, is a decay lemma in $\mathcal{F}$. Note, that lemma 1 does not immediately give us the desired result. The reason for stating different decay lemmas, is that lemma 1 works under less restrictive assumptions on the parameter sequence. We now give the adapted proof of the decay lemma, where we set $g^{(t)} := (p^{(t)} - \theta^{(t)})/\lambda^{(t)}$. With this convention, we see directly that

$$\begin{aligned}\mathcal{F}(\theta^{(t+1)}, g^{(t)}) &= \mathcal{L}(\theta^{(t+1)}) + \frac{\lambda^{(\infty)}}{\tau}|\theta^{(t+1)}|_1 - \frac{\lambda^{(\infty)}}{\tau}\langle\theta^{(t+1)}, g^{(t)}\rangle \\ &= \mathcal{L}(\theta^{(t+1)}) + \frac{\lambda^{(\infty)}}{\tau}D_{|\cdot|_1}^{g^{(t)}}(\theta^{(t+1)}, \theta^{(t)}).\end{aligned}$$

For notational ease we also set $\xi^{(t+1)} = (\theta^{(t+1)}, g^{(t)})$ and now state the desired decay in $\mathcal{F}$.

Lemma 6. *Let assumption 1 hold true, assume* (15) *and that* $\tau < \frac{2}{L}$*, then we have that*

$$\mathcal{F}(\xi^{(t+1)}) + \rho_1 |\theta^{(t+1)} - \theta^{(t)}|_2^2 \leq \mathcal{F}(\xi^{(t)}) + e^{(t)}$$

where $\rho_1 = \frac{1}{\tau} - \frac{L}{2}$ *and*

$$e^{(t)} := \frac{|\lambda^{(t+1)} - \lambda^{(\infty)}| + |\lambda^{(t)} - \lambda^{(\infty)}|}{\tau} |\theta^{(t+1)} - \theta^{(t)}|_1.$$

Proof. We begin as in the proof of lemma 1 to obtain

$$\begin{aligned}\mathcal{L}(\theta^{(t+1)}) + \rho_1 |\theta^{(t+1)} - \theta^{(t)}|_2^2 &\leq \mathcal{L}(\theta^{(t)}) - \frac{1}{\tau}\langle \lambda^{(t+1)} g^{(t+1)} - \lambda^{(t)} g^{(t)}, \theta^{(t+1)} - \theta^{(t)} \rangle \\ &= \mathcal{L}(\theta^{(t)}) - \frac{\lambda^{(\infty)}}{\tau}\langle g^{(t+1)} - g^{(t)}, \theta^{(t+1)} - \theta^{(t)} \rangle + E^{(t)},\end{aligned}$$

where

$$E^{(t)} := -\frac{1}{\tau}\langle (\lambda^{(t+1)} - \lambda^{(\infty)}) g^{(t+1)} - (\lambda^{(t)} - \lambda^{(\infty)}) g^{(t)}, \theta^{(t+1)} - \theta^{(t)} \rangle.$$

On the left hand side, we want to produce $\mathcal{F}(\theta^{(t+1)}, g^{(t)})$, for which we move the term $\frac{\lambda^{(\infty)}}{\tau} D_{|\cdot|_1}^{g^{(t)}}(\theta^{(t+1)}, \theta^{(t)}) = \frac{\lambda^{(\infty)}}{\tau} \langle \theta^{(t+1)}, g^{(t+1)} - g^{(t)} \rangle$ to the left, which yields

$$\begin{aligned}\mathcal{F}(\theta^{(t+1)}, g^{(t)}) + \rho_1 |\theta^{(t+1)} - \theta^{(t)}|_2^2 &\leq \mathcal{L}(\theta^{(t)}) + \frac{\lambda^{(\infty)}}{\tau}\langle g^{(t+1)} - g^{(t)}, \theta^{(t)} \rangle + E^{(t)} \\ &= \mathcal{L}(\theta^{(t)}) - \frac{\lambda^{(\infty)}}{\tau} D_{|\cdot|_1}^{g^{(t+1)}}(\theta^{(t)}, \theta^{(t+1)}) + E^{(t)} \\ &\leq \mathcal{L}(\theta^{(t)}) + \frac{\lambda^{(\infty)}}{\tau} D_{|\cdot|_1}^{g^{(t-1)}}(\theta^{(t)}, \theta^{(t-1)}) + E^{(t)} \\ &= \mathcal{F}(\theta^{(t)}, g^{(t-1)}) + E^{(t)}.\end{aligned}$$

What is left, is to estimate $E^{(t)}$, which we do in the following. An application of Hölder's inequality yields

$$\begin{aligned}E^{(t)} &\leq \frac{1}{\tau} |(\lambda^{(t+1)} - \lambda^{(\infty)}) g^{(t+1)} - (\lambda^{(t)} - \lambda^{(\infty)}) g^{(t)}|_\infty \, |\theta^{(t+1)} - \theta^{(t)}|_1 \\ &\leq \frac{1}{\tau} \left(|\lambda^{(t+1)} - \lambda^{(\infty)}| + |\lambda^{(t)} - \lambda^{(\infty)}| \right) |\theta^{(t+1)} - \theta^{(t)}|_1 = e^{(t)}.\end{aligned}$$

□

As in Benning et al. (2021) we consider the set of possible limit points

$$\Xi = \left\{ (\theta^{(\infty)}, g^{(\infty)}) : \text{there exists a subsequence of } (\theta^{(t+1)}, g^{(t)}) \text{ converging to } (\theta^{(\infty)}, g^{(\infty)}) \right\}$$

for which we need to ensure that it is non-empty and compact. Of course, our goal later on is to show that it is a single-valued set.

Lemma 7. *Under the assumptions of lemma 4, the set* Ξ *is non-empty and compact and furthermore*

$$\lim_{t \to \infty} \operatorname{dist}(\xi^{(t+1)}, \Xi) = 0.$$

Proof. Due to lemma 4 we have that $(\xi^{(t)})_{t \in \mathbb{N}}$ is a bounded sequence in $\mathbb{R}^{2d}$ and thus it has at least one convergent subsequence, which implies that Ξ is non-empty. The compactness follows from its definition as a set of cluster points. The convergence of the distance follows similarly. Assume otherwise that there would be a subsequence and $\delta > 0, J \in \mathbb{N}$, such that $\operatorname{dist}(\xi^{(t_j+1)}, \Xi) > \delta$ for all $j > J$, then due to the boundedness of this subsequence, it has itself a converging subsequence, whose limit would then be in Ξ. This contradicts the assumption and thus $\operatorname{dist}(\xi^{(t_j+1)}, \Xi) > \delta$ for all $j > J$ is impossible. □

Beyond that as in (Benning et al., 2021, Lemma 4.6) we show that the functional values of $\mathcal{F}(\theta^{(t+1)}, g^{(t)})$ indeed converge to the value of $\mathcal{L}$ on the cluster points in Ξ.

Lemma 8. *Under the assumptions of lemma 4, let $\xi^{(\infty)} \in \Xi$ then we have that*

$$\lim_{t\to\infty} \mathcal{F}(\xi^{(t+1)}) = \mathcal{F}(\xi^{(\infty)}) = \mathcal{L}(\theta^{(\infty)}).$$

Proof. Let $(t_j)_{j\in\mathbb{N}}$ be a subsequence, such that

$$\lim_{j\to\infty} (\theta^{(t_j+1)}, g^{(t_j)}) = (\theta^{(\infty)}, g^{(\infty)})$$

then we have that

$$\lim_{j\to\infty} \mathcal{F}(\theta^{(t_j+1)}, g^{(t_j)}) = \lim_{j\to\infty} \mathcal{L}(\theta^{(t_j+1)}) + \frac{\lambda^{(\infty)}}{\tau} D_{|\cdot|_1}^{g^{(t_j)}}(\theta^{(t_j+1)}, \theta^{(t_j)}) = \mathcal{L}(\theta^{(\infty)})$$

where we used the continuity of $\mathcal{L}$ and furthermore,

$$\begin{aligned} D_{|\cdot|_1}^{g^{(t_j)}}(\theta^{(t_j+1)}, \theta^{(t_j)}) &\leq \langle \theta^{(t_j+1)} - \theta^{(t_j)}, g^{(t_j+1)} - g^{(t_j)} \rangle \\ &\leq |g^{(t_j+1)} - g^{(t_j)}|_\infty \, |\theta^{(t_j+1)} - \theta^{(t_j)}|_1 \leq 2\sqrt{d}\, |\theta^{(t_j+1)} - \theta^{(t_j)}|_2 \xrightarrow{j\to\infty} 0 \end{aligned}$$

where the last limit follows already from lemma 4. Beyond that lemma 4 already established that $\mathcal{L}(\theta^{(t)})$ is a convergent sequence and therefore, also the whole sequence $(\mathcal{F}(\theta^{(t+1)}, g^{(t)})_{t\in\mathbb{N}}$ must converge to the same value. In particular, this shows that $\mathcal{F}$ is constant on Ξ, namely there is a single $F \in \mathbb{R}$ such that

$$F = \mathcal{F}(\xi^{(\infty)}) = \mathcal{L}(\theta^{(\infty)}) \qquad \text{for all } \xi^{(\infty)} \in \Xi.$$

□

We also need to control how the subdifferential of the surrogate evolves during the iteration. From Benning et al. (2021) we know that

$$\partial\mathcal{F}(\xi^{(t+1)}) \ni \begin{pmatrix} \nabla\mathcal{L}(\theta^{(t+1)}) + \frac{\lambda^{(\infty)}}{\tau}(g^{(t+1)} - g^{(t)}) \\ \frac{\lambda^{(\infty)}}{\tau}(\theta^{(t)} - \theta^{(t+1)}) \end{pmatrix} =: G^{(t+1)}$$

which allows to show the following bound on how far the subdifferential is from 0.

Lemma 9. *Let assumption 1 hold true and assume* (15)*, then*

$$\begin{aligned} &\operatorname{dist}(0, \partial\mathcal{F}(\xi^{(t+1)})) \leq |G^{(t+1)}|_2 \\ &\leq \left(L + \frac{1+\lambda^{(\infty)}}{\tau}\right) |\theta^{(t+1)} - \theta^{(t)}|_2 + \frac{\sqrt{d}}{\tau}(|\lambda^{(t)} - \lambda^{(\infty)}| + |\lambda^{(t+1)} - \lambda^{(\infty)}|). \end{aligned}$$

Proof. Following the proof of (Benning et al., 2021, lemma 4.3) we first observe that

$$|G^{(t+1)}|_2 \leq \left|\nabla\mathcal{L}(\theta^{(t+1)}) + \frac{\lambda^{(\infty)}}{\tau}(g^{(t+1)} - g^{(t)})\right|_2 + \frac{\lambda^{(\infty)}}{\tau}|\theta^{(t)} - \theta^{(t+1)}|_2.$$

Using the L-smoothness and plugging in the recursion, we can estimate the first term as

$$\begin{aligned} &|\nabla\mathcal{L}(\theta^{(t+1)}) + \frac{\lambda^{(\infty)}}{\tau}(g^{(t+1)} - g^{(t)})|_2 \leq L|\theta^{(t+1)} - \theta^{(t)}|_2 \\ &\qquad + |\nabla\mathcal{L}(\theta^{(t)}) + \frac{\lambda^{(\infty)}}{\tau}(g^{(t+1)} - g^{(t)})|_2 \\ &= L|\theta^{(t+1)} - \theta^{(t)}|_2 + \frac{1}{\tau}|\theta^{(t)} - \theta^{(t+1)} + \lambda^{(t)}g^{(t)} - \lambda^{(t+1)}g^{(t+1)} + \lambda^{(\infty)}(g^{(t+1)} - g^{(t)})|_2 \\ &\leq \left(L + \frac{1}{\tau}\right)|\theta^{(t)} - \theta^{(t+1)}|_2 + \frac{1}{\tau}|(\lambda^{(t)} - \lambda^{(\infty)})g^{(t)} + (\lambda^{(\infty)} - \lambda^{(t+1)})g^{(t+1)}|_2 \\ &\leq \left(L + \frac{1}{\tau}\right)|\theta^{(t)} - \theta^{(t+1)}|_2 + \frac{\sqrt{d}}{\tau}(|\lambda^{(t)} - \lambda^{(\infty)}| + |\lambda^{(t+1)} - \lambda^{(\infty)}|). \end{aligned}$$

Collecting the terms yields the desired result. □

We can now finally show the desired finite length property for the sequence $\theta^{(t)}$ which will then imply convergence. To do so, we employ a Kurdyka–Łojasiewicz type property on the surrogate functional $\mathcal{F}$, as in Benning et al. (2021). Typically, this assumption is stated with arbitrary concave functions φ. Unfortunately, the proof strategy we employ below requires us to additionally assume that $1/\varphi'$ is sub-additive. A specific class of concave functions that has this property are power functions $\varphi(y) = cy^{1-\alpha}$ where $\alpha \in [0, 1)$. A similar assumption is for example discussed in Frankel et al. (2015). We thus will make the following more restrictive KŁ-assumption:

Assumption 2. *We assume that for each $\bar{\xi} \in \operatorname{dom}(\mathcal{F})$ there are constants $\alpha_{\bar{\xi}} \in [0, 1), c_{\bar{\xi}}, \delta_{\bar{\xi}}, \eta_{\bar{\xi}} > 0$, such that for all $\xi \in B_{\delta_{\bar{\xi}}}(\bar{\xi}) \cap \{\xi : \mathcal{F}(\bar{\xi}) < \mathcal{F}(\xi) < \mathcal{F}(\bar{\xi}) + \eta_{\bar{\xi}}\}$, we have for $\varphi_{\bar{\xi}} = c_{\bar{\xi}}(1-\alpha_{\bar{\xi}})^{-1} y^{1-\alpha_{\bar{\xi}}}$*

$$\frac{c_{\bar{\xi}}}{\varphi'_{\bar{\xi}}(\mathcal{F}(\xi) - \mathcal{F}(\bar{\xi}))} = \big(\mathcal{F}(\xi) - \mathcal{F}(\bar{\xi})\big)^{\alpha_{\bar{\xi}}} \leq c_{\bar{\xi}} \operatorname{dist}(0, \partial \mathcal{F}(\xi)). \tag{KL}$$

Remark 3. *The above condition can for example be ensured if $\mathcal{L}$ is semi-algebraic, see for example Attouch et al. (2010).*

In assumption 2 the occurring constants are dependent on the point $\bar{\xi}$. In the subsequent proof, we want to apply this assumption for $\bar{\xi} \in \Xi$ and here it would be inconvenient to carry this dependence. To this end, (Bolte et al., 2014, Lemma 6) provides a uniform condition, which we state in the following:

Lemma 10. *(Bolte et al., 2014, Lemma 6) Let $\tilde{\Xi} \subset \operatorname{dom}(\mathcal{F})$ be a compact set, and assume that there is $F \in \mathbb{R}$ such that $\mathcal{F}(\bar{\xi}) = F \in \mathbb{R}$ for all $\bar{\xi} \in \tilde{\Xi}$, then there exist $\eta_2, c_{\tilde{\Xi}} > 0$, $\alpha_{\tilde{\Xi}} \in [0, 1)$ and $0 < \eta_1 < 1$ and a (single) function $\varphi(y) = c_{\tilde{\Xi}}(1 - \alpha_{\tilde{\Xi}})^{-1} y^{1-\alpha_{\tilde{\Xi}}}$ with $c_{\tilde{\Xi}} > 0$ and $\alpha_{\tilde{\Xi}} \in [0, 1)$ such that* (KL) *holds for all*

$$\xi \in \Big\{\tilde{\xi} : \operatorname{dist}(\tilde{\xi}, \tilde{\Xi}) < \eta_2, F < \mathcal{F}(\tilde{\xi}) < F + \eta_1\Big\} =: \mathcal{N}.$$

Proof. The proof is given in (Bolte et al., 2014, Lemma 6) for general concave φ. One can adapt the same covering argument to specialize the result to the power functions φ. □

In the following proof, we will apply this lemma for the set Ξ in lemma 7. Denote by $\varphi_{\Xi}(y) = c_{\Xi} y^{1-\alpha_{\Xi}}$ the function from the above lemma. For the finite length property, we require the convergence speed of $\lambda^{(t)}$ to be compatible with the exponent of the employed concave function φ. First of all we assume that $|\lambda^{(t)} - \lambda^{(\infty)}| \leq C_\lambda t^q$ and then, as will be clear from our argument below, we need

$$q < -\frac{1}{\alpha} - \frac{1}{2}.$$

To this end we assume $q < -\frac{3}{2}$ and choose $1 > \alpha > \max\{\alpha_{\Xi}, \frac{1}{-q-1/2}\}$, which ensures this condition. Furthermore, since $\alpha_{\Xi} \leq \alpha$ the KŁ-property also holds with α on $\mathcal{N}$ and from now on we will work with the following concave function:

$$\varphi(y) = c\, y^{1-\alpha} = \frac{c_{\Xi}}{(1-\alpha)} y^{1-\alpha}$$

The important property employed below is the following: for $a, b > 0$ we have

$$\frac{1}{\varphi'(a+b)} = \frac{(a+b)^\alpha}{c(1-\alpha)} \leq \frac{a^\alpha + b^\alpha}{c(1-\alpha)} = \frac{1}{\varphi'(a)} + \frac{1}{\varphi'(b)}$$

Having collected these ingredients we can now adapt the proof of Benning et al. (2021) for the finite length property.

Lemma 11. *(Finite length property) Let assumption 1 and 2 hold true, $\tau < \frac{2}{L}$ and that $|\lambda^{(t)} - \lambda^{(\infty)}| \leq C_\lambda t^q$ with $q < -3/2$, then we have that*

$$\sum_{t=0}^{\infty} |\theta^{(t+1)} - \theta^{(t)}|_2 < \infty.$$

Proof. The proof follows (Bolte et al., 2014, Theorem 1) and the adaptions from (Benning et al., 2021, Theorem 4.9) with the additional modifications introduced in our framework. We again use the abbreviation $\xi^{(t+1)} = (\theta^{(t+1)}, g^{(t)})$.

Let $(t_j)_{j\in\mathbb{N}}$ be a subsequence from lemma 4 such that

$$\lim_{j\to\infty} \theta^{(t_j)} = \theta^{(\infty)},$$

then lemma 8 implies $\lim_{t\to\infty} \mathcal{F}(\xi^{(t)}) = \mathcal{L}(\theta^{(\infty)})$. Compared to Benning et al. (2021) we have the additional difficulty that the error in lemma 6 could allow a situation where $\mathcal{F}(\xi^{(t+1)}) < \mathcal{L}(\theta^{(\infty)})$, thus it does not suffice to consider the difference $\mathcal{F}(\xi^{(t+1)}) - \mathcal{L}(\theta^{(\infty)})$ as in their proof. Instead, we consider the sequence

$$r^{(t+1)} := \mathcal{F}(\xi^{(t+1)}) + \sum_{k=t+1}^{\infty} e^{(k)} - \mathcal{L}(\theta^{(\infty)})$$

where we know from lemma 4 that $\sum_{k=0}^{\infty} e^{(k)} < \infty$ if $(\lambda^{(t)} - \lambda^{(\infty)})_{t\in\mathbb{N}} \in \ell^2(\mathbb{R})$. Then lemma 6 yields

$$r^{(t+1)} \leq \mathcal{F}(\xi^{(t)}) + e^{(t)} + \sum_{k=t+1}^{\infty} e^{(k)} - \mathcal{L}(\theta^{(\infty)}) = r^{(t)}$$

and thus $(r^{(t)})_{t\in\mathbb{N}}$ is a non-increasing null sequence and therefore bounded from below by 0. Furthermore due to lemma 6 we have

$$r^{(t)} - r^{(t+1)} = \mathcal{F}(\xi^{(t)}) - \mathcal{F}(\xi^{(t+1)}) + e^{(t)} \geq \rho_1 |\theta^{(t+1)} - \theta^{(t)}|_2^2, \tag{16}$$

which implies $\theta^{(t+1)} = \theta^{(t)}$ whenever $r^{(t+1)} = r^{(t)}$. If $r^{(T)} = 0$ for some finite $T \in \mathbb{N}$ we must have $r^{(t)} = 0$ for all $t \geq T$ and then the iteration is stationary from T on and there is nothing to show. We thus assume $r^{(t)} > 0$ for all $t \in \mathbb{N}$ from here on. We now aim to show a bound for the difference $|\theta^{(t+1)} - \theta^{(t)}|_2$ which is summable. If $\theta^{(t+1)} = \theta^{(t)}$ there is nothing to show and we thus only focus on the cases where $r^{(t+1)} < r^{(t)}$, which we assume from here on. Furthermore, we can choose T_η large enough such that

$$\left.\begin{aligned} \mathcal{F}(\xi^{(t)}) < \mathcal{L}(\theta^{(\infty)}) + \eta_1 \\ \operatorname{dist}(\xi^{(t)}, \Xi) < \eta_2 \end{aligned}\right\} \qquad \text{for all } t \geq T_\eta,$$

and we only consider $t \geq T_\eta$ in the following. We can then employ the concavity of φ to see that

$$\varphi'(r^{(t)}) \leq \frac{\varphi(r^{(t)}) - \varphi(r^{(t+1)})}{r^{(t)} - r^{(t+1)}}$$

and thus with (16)

$$|\theta^{(t+1)} - \theta^{(t)}|_2^2 \leq \frac{\varphi(r^{(t)}) - \varphi(r^{(t+1)})}{\rho_1\, \varphi'(r^{(t)})}. \tag{17}$$

Again, due to the error in the decay lemma, we cannot simply employ the same uniform KŁ-property of (Benning et al., 2021, Lemma A.12). In order to still obtain a useful estimate for $\varphi'(r^{(t)})$ we need to make a case distinction:

First case $\mathcal{F}(\xi^{(t)}) \leq \mathcal{L}(\theta^{(\infty)})$: Due to the non-negativity of $r^{(t)}$ this yields

$$r^{(t)} \leq \sum_{k=t}^{\infty} e^{(k)} =: w^{(t)}$$

and the concavity further yields

$$\frac{1}{\varphi'(r^{(t)})} \leq \frac{1}{\varphi'\left(w^{(t)}\right)}.$$

Second case $\mathcal{F}(\xi^{(t)}) > \mathcal{L}(\theta^{(\infty)})$: Here, we can now employ the uniform version of the KŁ-property. To do so, we need to split away the error term, which is exactly where we use the sub-additivity of $\frac{1}{\varphi'}$:

$$\frac{1}{\varphi'(r^{(t)})} \leq \frac{1}{\varphi'(\mathcal{F}(\xi^{(t)}) - \mathcal{L}(\theta^{(\infty)}))} + \frac{1}{\varphi'\left(w^{(t)}\right)}$$

and then using the KŁ-property (KL)

$$\frac{1}{\varphi'(r^{(t)})} \leq \operatorname{dist}(0, \partial\mathcal{F}(\xi^{(t)})) + \frac{1}{\varphi'\left(w^{(t)}\right)}.$$

Using lemma 9 and the bound $|\lambda^{(t)} - \lambda^{(\infty)}| \leq C_\lambda t^q$ we then obtain

$$\frac{1}{\varphi'(r^{(t)})} \leq \underbrace{\left(L + \frac{1 + \lambda^{(\infty)}}{\tau}\right)}_{=:\rho_2} |\theta^{(t)} - \theta^{(t-1)}|_2 + \frac{2\sqrt{d}C_\lambda}{\tau}(t-1)^q + \frac{1}{\varphi'\left(w^{(t)}\right)}$$

and we highlight that this bound is also true in the first case, since the additional terms are non-negative. Plugging this back into (17) we obtain

$$|\theta^{(t+1)} - \theta^{(t)}|_2^2 \leq \frac{\rho_2}{\rho_1}(\varphi(r^{(t)}) - \varphi(r^{(t+1)}))\left(|\theta^{(t)} - \theta^{(t-1)}|_2 + \frac{1}{\rho_2}\, c^{(t)}\right)$$

where the additional error term is given as

$$c^{(t)} = \frac{2\sqrt{d}C_\lambda}{\tau}(t-1)^q + \frac{1}{\varphi'(w^{(t)})}.$$

Proceeding as in Benning et al. (2021) we take the square root, multiply by two and then use Young's inequality $2\sqrt{ab} \leq a + b$ to infer that

$$2|\theta^{(t+1)} - \theta^{(t)}|_2 \leq \frac{\rho_2}{\rho_1}(\varphi(r^{(t)}) - \varphi(r^{(t+1)})) + |\theta^{(t)} - \theta^{(t-1)}|_2 + \frac{c^{(t)}}{\rho_2}$$

and thus subtracting $|\theta^{(t+1)} - \theta^{(t)}|_2$ on both sides we get

$$|\theta^{(t+1)} - \theta^{(t)}|_2 \leq \frac{\rho_2}{\rho_1}(\varphi(r^{(t)}) - \varphi(r^{(t+1)})) + |\theta^{(t)} - \theta^{(t-1)}|_2 - |\theta^{(t+1)} - \theta^{(t)}|_2 + \frac{c^{(t)}}{\rho_2}.$$

Summing this inequality from T_η to some $T > T_\eta$, we get

$$\begin{aligned}\sum_{t=T_\eta}^{T} |\theta^{(t+1)} - \theta^{(t)}|_2 &\leq \frac{\rho_2}{\rho_1}(\varphi(r^{(T_\eta)}) - \varphi(r^{(T+1)})) + |\theta^{(T_\eta)} - \theta^{(T_\eta-1)}|_2 - |\theta^{(T+1)} - \theta^{(T)}|_2 \\ &\quad + \frac{1}{\rho_2}\sum_{t=T_\eta}^{T} c^{(t)} \\ &\leq \frac{\rho_2}{\rho_1}\varphi(r^{(T_\eta)}) + |\theta^{(T_\eta)} - \theta^{(T_\eta-1)}|_2 + \frac{1}{\rho_2}\sum_{t=T_\eta}^{T} c^{(t)}.\end{aligned}$$

The only thing left to show is that $(c^{(t)})_{t\in\mathbb{N}} \in \ell^1(\mathbb{R})$, which would then directly yield the desired finite length property from the above inequality. To establish this, we first observe that

$$\begin{aligned}\sum_{t=T_\eta}^{T} c^{(t)} &\leq \frac{2\sqrt{d}C_\lambda}{\tau}\sum_{t=T_\eta}^{T}(t-1)^q + \sum_{t=T_\eta}^{T}\frac{1}{\varphi'(w^{(t)})} \\ &\leq \frac{2\sqrt{d}C_\lambda}{\tau}\sum_{t=T_\eta}^{\infty}(t-1)^q + \sum_{t=T_\eta}^{T}\frac{1}{\varphi'(w^{(t)})},\end{aligned}$$

where the first series converges due to choice of q. Now for the second sum, we first estimate

$$w^{(t)} = \sum_{k=t}^{\infty} e^{(k)} \leq \frac{2\sqrt{d}C_\lambda}{\tau} \sum_{k=t}^{\infty} k^q |\theta^{(k+1)} - \theta^{(k)}|_2$$
$$\leq \frac{2\sqrt{d}\, C_\lambda}{\tau} \sqrt{\sum_{k=t}^{\infty} k^{2q}} \sqrt{\sum_{k=t}^{\infty} |\theta^{(k+1)} - \theta^{(k)}|_2^2},$$

where we employed the Cauchy–Schwarz inequality on the Hilbert space $\ell^2(\mathbb{R})$. Now from lemma 4 we already have that $\sum_{k=0}^{\infty} |\theta^{(k+1)} - \theta^{(k)}|_2^2 < \infty$ and thus

$$w^{(t)} \leq C \sqrt{\sum_{k=t}^{\infty} k^{2q}}, \qquad \text{with} \quad C = \frac{2\sqrt{d}C_\lambda}{\tau} \sqrt{\sum_{k=0}^{\infty} |\theta^{(k+1)} - \theta^{(k)}|_2^2}.$$

Since φ is concave, $\frac{1}{\varphi'}$ is non-decreasing and thus

$$\frac{1}{\varphi'(w^{(t)})} \leq \frac{1}{\varphi'\left(C\sqrt{\sum_{k=t}^{\infty} k^{2q}}\right)} = (*).$$

Now using the specific form of $\varphi(y) = cy^{1-\alpha}$ we get

$$(*) = \frac{C^\alpha}{c\,(1-\alpha)} \left(\sum_{k=t}^{\infty} k^{2q}\right)^{\alpha/2}$$

In order to estimate the above sum, we use an comparison with integral to derive

$$\sum_{k=t}^{\infty} k^{2q} \leq t^{2q} + \int_t^{\infty} \tilde{t}^{2q} \mathrm{d}\tilde{t} = t^{2q} - \frac{t^{2q+1}}{2q+1} = \frac{(2q+1)t^{2q} - t^{2q+1}}{2q+1} \leq \frac{2qt^{2q+1}}{2q+1}.$$

Therefore, we obtain

$$\sum_{t=T_\eta}^{\infty} \frac{1}{\varphi'(w^{(t)})} \leq \frac{c}{1-\alpha} \left(\frac{2qC^2}{2q+1}\right)^{\alpha/2} \sum_{t=1}^{\infty} t^{(2q+1)\alpha/2}.$$

Now we use the assumption

$$q < -\frac{1}{\alpha} - \frac{1}{2}$$

to infer that the sum is finite. This closes the proof. □

With this we can now give the proof of proposition 1:

Proof. Let $\epsilon > 0$ be given, then since $\theta^{(t)}$ has finite length there exists T such that for all $\tilde{t} \geq t \geq T$ we have

$$\sum_{k=t}^{\tilde{t}} |\theta^{(k+1)} - \theta^{(k)}|_2 < \epsilon.$$

Now, for any $\tilde{t} > t \geq T$ this yields

$$|\theta^{(\tilde{t})} - \theta^{(t)}|_2 = \left|\sum_{k=t}^{\tilde{t}-1} \theta^{(k+1)} - \theta^{(k)}\right|_2 \leq \sum_{k=t}^{\tilde{t}-1} \left|\theta^{(k+1)} - \theta^{(k)}\right|_2 < \epsilon.$$

Since $\epsilon > 0$ was arbitrary $(\theta^{(t)})_{t\in\mathbb{N}}$ is a Cauchy-sequence in $\mathbb{R}^d$ and thus convergent with limit $\theta^{(\infty)} \in \mathbb{R}^d$. Due to lemma 5 this implies $\nabla\mathcal{L}(\theta^{(\infty)}) = 0$. □

A.3 PROOF OF LEMMA 3

Proof. By the proximal step and the definition of $m_i^{(t)}$,

$$\theta_i^{(t)} = \operatorname{sign}\big(p_i^{(t)}\big)\max\big(\big|p_i^{(t)}\big| - \lambda^{(t)},\, 0\big) = \begin{cases} p_i^{(t)} - \lambda^{(t)}\operatorname{sign}\big(p_i^{(t)}\big) & \text{if } \big|p_i^{(t)}\big| > \lambda^{(t)}, \\ 0 & \text{else}, \end{cases}$$
$$= m_i^{(t)}\big(p_i^{(t)} - \lambda^{(t)}\operatorname{sign}\big(p_i^{(t)}\big)\big),$$

which proves (a). For (b), $\big|\theta_i^{(t)}\big| = \big|p_i^{(t)}\big| - \lambda^{(t)}$ whenever $m_i^{(t)} = 1$. Hence, for any i, j with $m_i^{(t)} = m_j^{(t)} = 1$, we have $\big|\theta_i^{(t)}\big| > \big|\theta_j^{(t)}\big|$ if and only if $\big|p_i^{(t)}\big| > \big|p_j^{(t)}\big|$, which proves (b). □

A.4 PROOFS OF PROPOSITION 2

Proof. **Part (i)**: Substituting the dual step in Lemma 3 (a) yields

$$\big|\theta_i^{(t+1)}\big| = \begin{cases} \big|p_i^{(t+1)}\big| - \lambda^{(t+1)} = \big|p_i^{(0)} - \tau\sum_{u=0}^{t}\nabla_{\theta_i}\mathcal{L}(\theta^{(u)})\big| - \lambda^{(t+1)}, & \big|p_i^{(t+1)}\big| > \lambda^{(t+1)}, \\ 0, & \text{else}. \end{cases}$$

Part (ii): The assumption is that the product of $m_i^{(t)}\operatorname{sign}(p_i^{(t)})$ at two consecutive iterations is equal. There are three possible values of this product: 0 or ± 1.

Case $m_i^{(t)}\operatorname{sign}(p_i^{(t)}) = m_i^{(t+1)}\operatorname{sign}(p_i^{(t+1)}) = 0$: Lemma 3(a) implies that $\theta_i^{(t)} = \theta_i^{(t+1)} = 0$, which holds for both cases in (10) with $m_i^{(t)} = 0$.

Case $m_i^{(t)}\operatorname{sign}(p_i^{(t)}) = m_i^{(t+1)}\operatorname{sign}(p_i^{(t+1)}) = \pm 1$: Both masks are 1, thus $\operatorname{sign}\big(p_i^{(t)}\big) = \operatorname{sign}\big(p_i^{(t+1)}\big)$. Writing the two iterates of Lemma 3(a) and using $\operatorname{sign}(p_i^{(t)}) = \operatorname{sign}(p_i^{(t+1)})$ gives

$$\theta_i^{(t+1)} = p_i^{(t+1)} - \lambda^{(t+1)}\operatorname{sign}(p_i^{(t)}),$$
$$\theta_i^{(t)} = p_i^{(t)} - \lambda^{(t)}\operatorname{sign}(p_i^{(t)})$$
$$= \operatorname{sign}\big(p_i^{(t)}\big)\big(\big|p_i^{(t)}\big| - \lambda^{(t)}\big)$$
$$\implies \operatorname{sign}\big(p_i^{(t)}\big) = \operatorname{sign}\big(\theta_i^{(t)}\big) \qquad \big(\big|p_i^{(t)}\big| - \lambda^{(t)} > 0\big).$$

Subtracting step t from step $t{+}1$, and using $p_i^{(t+1)} - p_i^{(t)} = -\tau\nabla_{\theta_i}\mathcal{L}(\theta^{(t)})$ from the dual update and $\operatorname{sign}\big(p_i^{(t)}\big) = \operatorname{sign}\big(\theta_i^{(t)}\big)$

$$\theta_i^{(t+1)} - \theta_i^{(t)} = -\tau\nabla_{\theta_i}\mathcal{L}(\theta^{(t)}) - \operatorname{sign}(\theta_i^{(t)})\big(\lambda^{(t+1)} - \lambda^{(t)}\big),$$
$$\theta_i^{(t+1)} = \theta_i^{(t)} - \tau\nabla_{\theta_i}\mathcal{L}(\theta^{(t)}) - \operatorname{sign}(\theta_i^{(t)})\big(\lambda^{(t+1)} - \lambda^{(t)}\big).$$

Finally, we rewrite both cases of $m_i^{(t)} \in \{0,\, 1\}$ with a single expression as

$$\theta_i^{(t+1)} = m_i^{(t)}\big(\theta_i^{(t)} - \tau\nabla_{\theta_i}\mathcal{L}(\theta^{(t)}) - \operatorname{sign}(\theta_i^{(t)})(\lambda^{(t+1)} - \lambda^{(t)})\big),$$

which is (10). By construction, (8) holds $\lambda^{(t)}$ constant for $\Delta t - 1$ out of Δt steps, forcing the last term to 0, rendering the freezing steps equal to the masked SGD, which proves the claim in (ii).

Part (iii): The assumption here is that $m_i^{(t)}\operatorname{sign}(p_i^{(t)}) \neq m_i^{(t+1)}\operatorname{sign}(p_i^{(t+1)})$. If $m_i^{(t)} = m_i^{(t+1)} = 0$, both products are 0 and thus equal. Therefore, at least one of $m_i^{(t)}$ and $m_i^{(t+1)}$ equals 1. The lower bound is straightforward and holds by the definition of the proximal step

$$|\theta_i^{(s)}| = |p_i^{(s)}| - \lambda^{(s)} > 0 \qquad \text{for } m_i^{(s)} = 1,\ s \in \{t,\, t{+}1\},$$

hence $\max(|\theta_i^{(t)}|, |\theta_i^{(t+1)}|) > 0$. To derive the upper bound, we study the following cases separately.

Case $m_i^{(t)} \neq m_i^{(t+1)}$: $m_i^{(t)}=0$ or $m_i^{(t+1)}=0$. Take the case $m_i^{(t)}=0 \implies m_i^{(t+1)}=1$, with $\theta_i^{(t)}=0$ and $\left|p_i^{(t)}\right| \leq \lambda^{(t)}$. The dual step and Lemma 3(a) bound $\theta_i^{(t+1)}$ as follows

$$\begin{aligned} p_i^{(t+1)} &= p_i^{(t)} - \tau \nabla_{\theta_i} \mathcal{L}(\theta^{(t)}) \\ \left|p_i^{(t+1)}\right| &\leq \left|p_i^{(t)}\right| + \tau \left|\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})\right| \\ &\leq \lambda^{(t)} + \tau \left|\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})\right| \\ \left|\theta_i^{(t+1)}\right| &= \left|p_i^{(t+1)}\right| - \lambda^{(t+1)} \\ &\leq \lambda^{(t)} - \lambda^{(t+1)} + \tau \left|\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})\right| . \end{aligned}$$

The opposite direction exchanges t and $t+1$, and it bounds $\theta_i^{(t)}$ as

$$\left|\theta_i^{(t)}\right| \leq \lambda^{(t+1)} - \lambda^{(t)} + \tau |\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})|.$$

Therefore, both $\theta_i^{(t)}$ and $\theta_i^{(t+1)}$ are bounded by

$$\max(\left|\theta_i^{(t)}\right|, \left|\theta_i^{(t+1)}\right|) \leq |\lambda^{(t+1)} - \lambda^{(t)}| + \tau |\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})|.$$

Case $m_i^{(t)}=m_i^{(t+1)}=1$ *and* $\operatorname{sign}(p_i^{(t)}) \neq \operatorname{sign}(p_i^{(t+1)})$: The two subgradients lie on opposite sides of 0, therefore their distance is the sum of their magnitudes,

$$\begin{aligned} \left|p_i^{(t)}\right| + \left|p_i^{(t+1)}\right| &= \left|p_i^{(t+1)} - p_i^{(t)}\right| = \tau \left|\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})\right| \\ &\implies \left|\theta_i^{(t)}\right| + \left|\theta_i^{(t+1)}\right| = \tau \left|\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})\right| - (\lambda^{(t)} + \lambda^{(t+1)}) \\ &\implies \max(\left|\theta_i^{(t)}\right|, \left|\theta_i^{(t+1)}\right|) < \tau \left|\nabla_{\theta_i} \mathcal{L}(\theta^{(t)})\right| \end{aligned}$$

The last statement holds since both magnitudes are positive, therefore each is smaller than the sum. Therefore both cases under the assumption satisfy (11), which proves (iii).

Part (iv): At $t \bmod \Delta t=0$, (8) sets $\lambda^{(t+1)}=\left|p^{(t+1)}\right|_{[d-\mathsf{k}^*]}$, where $\left|p^{(t+1)}\right|_{[1]} \leq \cdots \leq \left|p^{(t+1)}\right|_{[d]}$ denote the magnitudes of $p^{(t+1)}$ sorted in ascending order. The no-tie assumption $\left|p^{(t+1)}\right|_{[d-\mathsf{k}^*+1]} > \left|p^{(t+1)}\right|_{[d-\mathsf{k}^*]}$ places the k^* largest magnitudes strictly above $\lambda^{(t+1)}$, and the other $d-\mathsf{k}^*$ magnitudes at or below it. Concretely,

$$\begin{aligned} m_i^{(t+1)} &= \mathbf{1}\{\left|p_i^{(t+1)}\right| > \lambda^{(t+1)}\} \\ &= \mathbf{1}\{\left|p_i^{(t+1)}\right| > \left|p^{(t+1)}\right|_{[d-\mathsf{k}^*]}\} \\ &\implies \operatorname{Card}\big(\{i : m_i^{(t+1)} = 1\}\big) = \mathsf{k}^* \\ &\implies \left|\theta^{(t+1)}\right|_0 = \left|m^{(t+1)} \odot p^{(t+1)}\right|_0 = \mathsf{k}^* \\ &\implies \mathsf{s}\big(\theta^{(t+1)}\big) = 1 - \frac{\left|\theta^{(t+1)}\right|_0}{d} = 1 - \frac{\mathsf{k}^*}{d} = \mathsf{s}^*, \end{aligned}$$

which proves (iv). The last line is by definition in (2). □

B SUBGRADIENT CORRECTION

We comment on the possibility of correcting $p^{(t+1)}$ in (6) such that it remains a valid subgradient at the next iteration. We consider the elastic net functional

$$\phi_{\lambda^{(t)}} = \frac{1}{2}|\cdot|_2^2 + \lambda^{(t)}|\cdot|_1.$$

Now, if $p^{(t)} \in \partial\phi_{\lambda^{(t)}}(\theta)$ for some $\lambda^{(t)} > 0$, for a different $\tilde{\lambda}^{(t)} > 0$ we have

$$\begin{aligned} p^{(t)} \in \theta + \lambda^{(t)}\partial|\cdot|_1(\theta) &\Rightarrow \frac{p^{(t)} - \theta}{\lambda^{(t)}} \in \partial|\cdot|_1(\theta) \\ &\Rightarrow \frac{\tilde{\lambda}^{(t)}}{\lambda^{(t)}}(p^{(t)} - \theta) + \theta \in \theta + \tilde{\lambda}^{(t)}\partial|\cdot|_1(\theta) \\ &\Rightarrow \frac{\tilde{\lambda}^{(t)}}{\lambda^{(t)}}\, p^{(t)} + \left(1 - \frac{\tilde{\lambda}^{(t)}}{\lambda^{(t)}}\right)\theta \in \partial\phi_{\tilde{\lambda}^{(t)}}(\theta). \end{aligned}$$

At indices i where $\theta_i^{(t)} = 0$, we can simply clip $p^{(t)}$ to the range $[-\tilde{\lambda}^{(t)}, \tilde{\lambda}^{(t)}]$ which yields the subgradient correction $C(p, \theta, \lambda, \tilde{\lambda}) \in \mathbb{R}^d$ with

$$C(p^{(t)}, \theta^{(t)}, \lambda^{(t)}, \tilde{\lambda}^{(t)})_i := \begin{cases} \frac{\tilde{\lambda}^{(t)}}{\lambda^{(t)}}\, p_i^{(t)} + \left(1 - \frac{\tilde{\lambda}^{(t)}}{\lambda^{(t)}}\right)\theta_i^{(t)} & \text{if } \theta_i^{(t)} \neq 0, \\ \min\{\max\{p_i^{(t)}, -\tilde{\lambda}^{(t)}\}, \tilde{\lambda}^{(t)}\} & \text{else.} \end{cases}$$

This suggests modifying the update in (6) as follows

$$\begin{aligned} &\tilde{p}^{(t+1)} := p^{(t)} - \tau\nabla\mathcal{L}(\theta^{(t)}) \\ &\theta^{(t+1)} := \nabla(\phi_{\lambda^{(t)}})^*\left(\tilde{p}^{(t+1)}\right) && \Rightarrow \tilde{p}^{(t+1)} \in \partial\phi_{\lambda^{(t)}}(\theta^{(t+1)}) \\ &p^{(t+1)} := C(\tilde{p}^{(t+1)}, \theta^{(t+1)}, \lambda^{(t)}, \lambda^{(t+1)}) && \Rightarrow p^{(t+1)} \in \partial\phi_{\lambda^{(t+1)}}(\theta^{(t+1)}). \end{aligned}$$

Note that with the subgradient correction, $\lambda^{(t+1)}$ is updated based on $\tilde{p}^{(t+1)}$ instead of $p^{(t+1)}$ in (8). The sparsity guarantee in Proposition 2 (iv) then no longer holds. An intuitive explanation is the following. When $|\theta^{(t+1)}|_0 > \mathsf{k}^*$ at an adaptation step, $\lambda^{(t+1)}$ increases so that the soft-threshold sets more components of $\theta^{(t+2)}$ to zero. However, the rescaling by $\frac{\lambda^{(t+1)}}{\lambda^{(t)}}$ increases the magnitude of the subgradient of coordinates i with $\theta_i^{(t+1)} \neq 0$, so the new threshold $\lambda^{(t+1)}$ fails to prune these coordinates. The correction therefore partially negates the intended effect of the increase of λ. For this reason, we did not pursue this direction in the main results.

C PROPERTIES OF THE SIGNED GAP ALGORITHM

In the following, we study some properties of our heuristic SGap algorithm. From the update rule (12), the relative change is

$$\frac{|\lambda^{(t+1)} - \lambda^{(t)}|}{\lambda^{(t)}} = \left|(1 + \alpha^{C^{(t)}}|\epsilon^{(t)}|)^{\operatorname{sign}(\epsilon^{(t)})} - 1\right|.$$

For $\operatorname{sign}(\cdot) \in \{\pm 1\}$, the relative change is expanded as

$$\frac{|\lambda^{(t+1)} - \lambda^{(t)}|}{\lambda^{(t)}} = \begin{cases} (1 + \alpha^{C^{(t)}}|\epsilon^{(t)}|) - 1 \;=\; \alpha^{C^{(t)}}|\epsilon^{(t)}|, & \text{if } \mathsf{s}^* \geq \epsilon^{(t)} \geq 0 \quad (18) \\ \left|(1 + \alpha^{C^{(t)}}|\epsilon^{(t)}|)^{-1} - 1\right| \;=\; \dfrac{\alpha^{C^{(t)}}|\epsilon^{(t)}|}{1 + \alpha^{C^{(t)}}|\epsilon^{(t)}|}, & \text{if } 0 > \epsilon^{(t)} \geq \mathsf{s}^* - 1. \quad (19) \end{cases}$$

By choosing $\alpha > 0$, (18) always upper bounds (19). Thus, $0 \leq \alpha^{C^{(t)}}|\epsilon^{(t)}| \leq |\epsilon^{(t)}|$ is an upper bound. We note that $\epsilon^{(t)}$ also bounds the shift in the proximal threshold, which we define as $|\lambda^{(t+1)} - \lambda^{(t)}|$, by $|\epsilon^{(t)}|\lambda^{(t)}$. This shift bounds how far the proximal threshold moves between updates; the number of parameters crossing the threshold additionally depends on the local density of $\left|p_i^{(t)}\right|$ near $\lambda^{(t)}$. We also note that $\dfrac{\alpha^{C^{(t)}}|\epsilon^{(t)}|}{1 + \alpha^{C^{(t)}}|\epsilon^{(t)}|} \approx \alpha^{C^{(t)}}|\epsilon^{(t)}|$ for $\epsilon^{(t)} \approx 0$, which means asymmetry has minimal impact on the oscillation around the target.

Table 3: Data and optimization settings.

Setting	CIFAR-100	TinyImageNet	ImageNet-1K
Classes	100	200	1000
Input resolution	32×32	64×64	224×224
Train / validation split	90%/10%	90%/10%	98%/2%
Augmentation	random crop, horizontal flip		
Epochs	200	200	100
Batch size	128	128	256
Weight decay†	5×10^{-4}	5×10^{-4}	1×10^{-4}
Optimizer	SGD/LinBreg, momentum=0.9		
Learning rate scheduler	cosine annealing with linear warm-up over 10% of the epochs		
Learning rate (SGD / LinBreg)	0.01 (start) → 0.1 (base)		
Learning rate (AdaBreg / AdamW)	0.001 (start) → 0.01 (base)		

† For LinBreg, weight decay is applied to the dual variable rather than to the weights directly; its effect on the weights is filtered through the proximal operator.

D EXPERIMENTAL SETUP

Table 3 lists the dataset details and the complete optimization settings, and Table 4 summarizes the setup of our baseline methods. All models were trained from scratch. Non-MD models used cross-entropy loss and SGD optimizer with momentum=0.9. For MD, LinBreg was used with the same hyperparameters to allow direct comparison. Normalization layers and bias parameters were excluded from the sparse optimization. They were not pruned in the pruning case, and were trained via SGD when using LinBreg. All experiments were run on a single NVIDIA A100 or A40 GPU, and the results were reported over 3 random seeds.

Baselines. For the baselines, we used the best GraNet variant, with $\mathsf{s}(\theta^{(0)})=0.5$, and the original RigL settings. For the non-adaptive (original) LinBreg, sparsity cannot be explicitly controlled. As a result, the same λ leads to a different final sparsity depending on the dataset and model. We therefore tuned two different sets of λ, one for ImageNet-1K and one for the remaining datasets, such that the final sparsity roughly ranges from 90% to 99%. Since the final sparsity can vary, we do not force the condition (7) on LinBreg and instead report the results of the model with the highest validation accuracy, regardless of the actual sparsity it achieves.

FLOPs. We count FLOPs following Evci et al. (2020) and Lunk et al. (2026), to which we refer for details. The training cost per iteration depends on whether a method computes the gradient for the active weights only or for all weights. Magnitude pruning and SET compute it for the active weights only. RigL and GraNet do the same, but compute it for all weights every ΔT_{mask} iterations to update the mask. LinBreg and our methods compute it for all weights at every iteration. Training FLOPs are summed over all training samples, using the layer densities at the end of each epoch, and inference FLOPs correspond to a single forward pass on one input sample at the evaluated checkpoint. We report both relative to the dense network.

E EXPERIMENTAL RESULTS

E.1 METHOD VALIDATION

Reaching s^*: Here we show the evolution of sparsity and λ throughout training on TinyImageNet at different sparsity rates and models. In TopK, SGap and their sparsity-scheduled version, the target sparsity is reached and maintained. Note that the values $\lambda^{(t)}$ needed to maintain s^* changes as training progresses.

Convergence analysis: We show that λ adaptation does not degrade the convergence. Figure 4 shows the convergence of ResNet-50 on ImageNet-1K at 95% sparsity. The curves indicate that the adaptation seen in TopK and SGap results in *faster* convergence on the validation set relative

Table 4: Sparse training hyperparameters.

Ours	**Value**
ζ (7)	0.5%
Δt (8), (12)	50
α (12)	0.95
$\lambda^{(0)}$	0.01
$\mathsf{s}(\theta^{(0)})$	99%
Baselines	**Description**
LinBreg (non-adaptive)	$\lambda \in \{0.9,\ 1.1,\ 1.25\}$ for CIFAR-100/ TinyImageNet, and $\{2.0,\ 4.0\}$ for ImageNet-1K.
GraNet (Liu et al., 2021a)	$\mathsf{s}(\theta^{(0)})$=0.5 and Δt_{mask}=1000.
RigL (Evci et al., 2020)	ERK mask, Δt_{mask}=100, drop fraction 0.3, stopped at 75% of the training.
SET	Exactly RigL's settings (not the original). The difference is using random growth criterion.
GMP	$\mathsf{s}(\theta^{(0)})$=0, global magnitude pruning

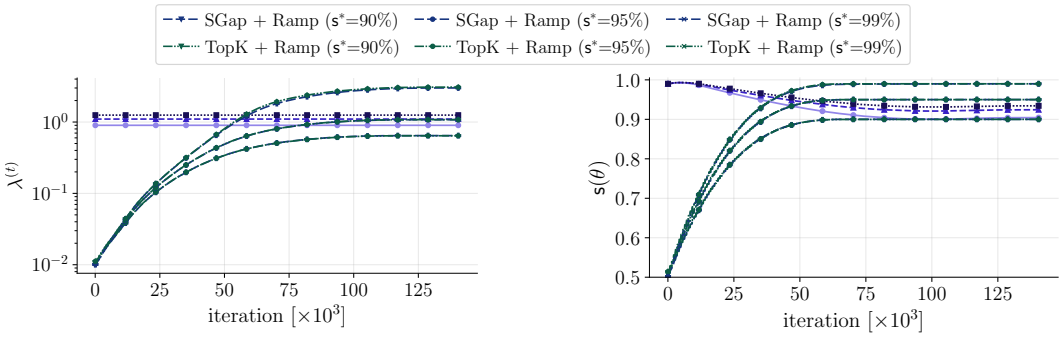


Figure 3: Profiles of $\mathsf{s}(\theta)$ and $\lambda^{(t)}$ of ResNet-50 on TinyImageNet when employing the sparsity schedule in (13) against fixed λ for reference.

to the fixed λ at early stages. We attribute this to the quicker reduction of sparsity (as visualized in Figure 1), thereby admitting more weights early, whereas fixing λ results in gradual sparsity drop.

Generalization to AdaBreg: So far, our analysis of TopK and SGap adaptations has been shown only for LinBreg steps. Here we show that both proposed adaptations generalize sparsity control to AdaBreg with Adam-style momenta (Bungert et al., 2022). Figures 5 and 6 show the $\mathsf{s}(\theta)$ / $\lambda^{(t)}$ profiles and convergence of ResNet-18 trained with AdaBreg on CIFAR-100, respectively, while

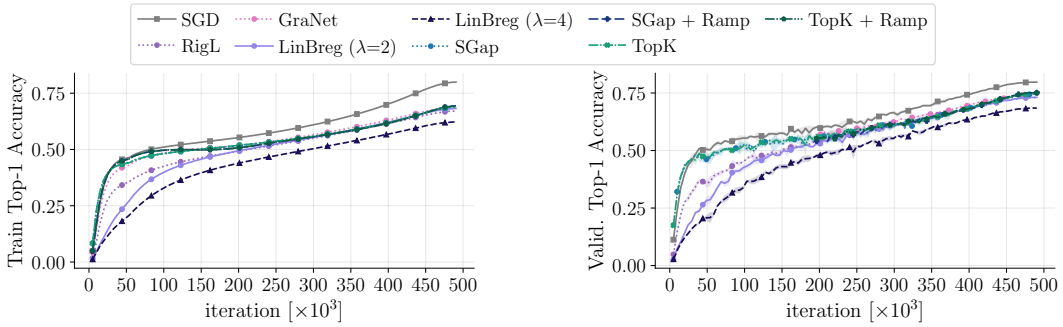


Figure 4: Convergence of ResNet-50 evaluated on ImageNet-1K at s^*=95%. Note that models are evaluated on the validation set only when they satisfy (7).

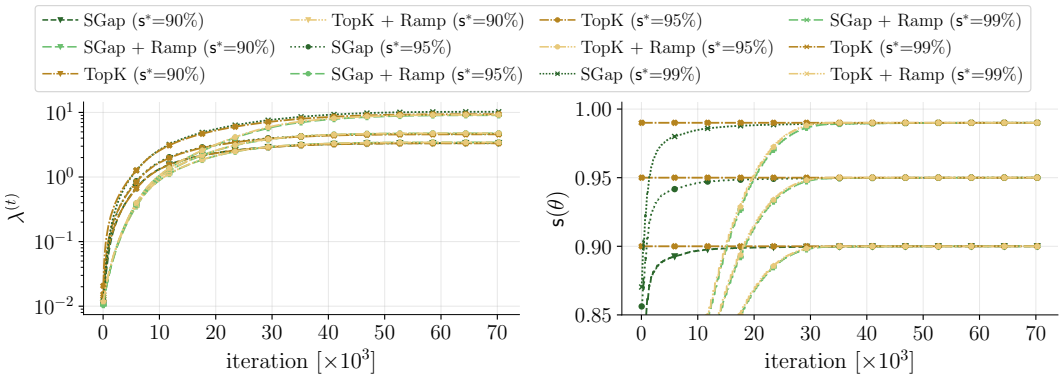


Figure 5: Evolution of $\mathsf{s}(\theta)$ and $\lambda^{(t)}$ measured on ResNet-18 evaluated on CIFAR-100 at different s^*.

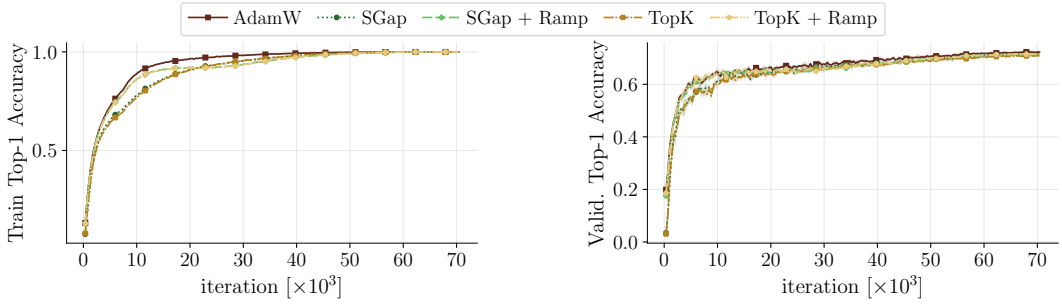


Figure 6: Convergence of ResNet-18 when trained using our adaptive AdaBreg variants at $\mathsf{s}^*{=}95\%$ versus AdamW on CIFAR-100.

Table 5 shows the top-1 accuracy scores. Figure 5 shows the sparsity target reached for every sparsity target and method using AdaBreg. This demonstrates that our methods reliably control sparsity for the Adam-style dual update step. The trajectory of $\lambda^{(t)}$ also follows a similar pattern, however, its values appear higher than its LinBreg counterparts.

At 90% sparsity, the top-1 test accuracy of both our AdaBreg is on-par with the dense model trained by AdamW (Loshchilov & Hutter, 2017). The TopK adaptation shows a sharp degradation at 99% sparsity, even compared to SGap. Using sparsity schedule appears to compensate for this degradation, restoring accuracy from only 65.7% to 70.8%. Apart from that case, both adaptation schemes are roughly on-par with one another.

E.2 ABLATION STUDY

Subgradient correction: We examine the effect of adapting the MD update step to apply the subgradient-correction scheme proposed in Appendix B. The results are reported in Figure 7 and Table 7 on CIFAR-100 and ResNet-18 for TopK and SGap adaptations. The sparsity and $\lambda^{(t)}$ follow nearly identical trajectories on both schemes. The notable exception is subgradient correction with TopK, where it does not satisfy the sparsity condition early in the training. However, both TopK and SGap reach s^* on all 4 variants.

The training accuracy of both subgradient-corrected variants is *higher* than those obtained by our baseline schemes. The generalization on the validation set remains similar across methods for most of the training duration, except towards the end, where subgradient-correction leads to slightly worse accuracies. The test accuracies also demonstrate the same generalization pattern: applying subgradient-correction results in slightly worse results across different sparsity levels and the two adaptation schemes.

Table 5: Top-1 accuracy [%] of AdaBreg of ResNet-18 on CIFAR-100 at different s^*. 'Ramp' refers to sparsity schedule. The best results per sparsity level are highlighted.

Method	90%	95%	99%
ResNet-18 (Dense, AdamW)		$72.0_{\pm 0.1}$	
AdaBreg + TopK	$71.5_{\pm 0.4}$	$70.5_{\pm 0.1}$	$65.7_{\pm 0.5}$
AdaBreg + SGap	$71.3_{\pm 0.2}$	$71.0_{\pm 0.3}$	$69.2_{\pm 0.3}$
AdaBreg + TopK + Ramp	$\mathbf{72.4_{\pm 0.1}}$	$\mathbf{72.0_{\pm 0.2}}$	$\mathbf{70.8_{\pm 0.2}}$
AdaBreg + SGap + Ramp	$72.2_{\pm 0.2}$	$71.8_{\pm 0.1}$	$70.6_{\pm 0.2}$

Table 6: Top-1 accuracy [%] over different Δt reported on ResNet-18 and CIFAR-100 at s^*=95%. Note that we use only models that satisfy (7), which explains the drop in accuracy at Δt=10000.

Method	Δt				
	1	50	250	1000	10000
TopK	$75.0_{\pm 0.16}$	$75.2_{\pm 0.3}$	$75.4_{\pm 0.2}$	$75.3_{\pm 0.4}$	$68.7_{\pm 0.3}$
SGap	$75.6_{\pm 0.4}$	$75.3_{\pm 0.1}$	$75.3_{\pm 0.4}$	$\mathbf{75.7_{\pm 0.1}}$	$69.0_{\pm 0.3}$

Influence of Δt: We study the effect of the hyperparameter Δt on convergence and the final model. In our framework, it defines the frequency of updating $\lambda^{(t)}$ (the mask in our case is generally updated per step). Despite the simple intuition behind it, this characterization may hide the other aspect related to Δt, namely: the size of *magnitude shift* $|\lambda^{(t+1)} - \lambda^{(t)}|$ needed to meet the sparsity constraint.

Equations (10) and (11) provide an insight into the effect of Δt. While larger values imply longer freezing steps (i.e., $\lambda^{(t+1)} = \lambda^{(t)}$), which are masked SGD steps, the adaptation step shifts the weights by the term $\operatorname{sign}(\theta^{(t)})(\lambda^{(t)} - \lambda^{(t+1)})$. The larger Δt, the larger the drift in the subgradients $|p^{(t+\Delta t)} - p^{(t)}|$, the bigger $|\lambda^{(t+1)} - \lambda^{(t)}|$. Larger values imply larger departure from the weights optimized by SGD to minimize the loss. Therefore, increasing Δt has the effect of introducing larger tension between converging to a minimum versus maintaining sparsity each time adaptation steps occur.

We investigate the influence of Δt on ResNet-18 and CIFAR-100. Our findings are reported in Table 6 and Figure 8. We found that the training loss behaves nearly identically for $\Delta t \in \{1, 50, 250\}$, while validation only slightly varies. However, the same curves show notably higher variance for larger Δt, especially on the validation set. The top-1 test accuracy seems robust for all choices but for Δt=10000. The sparsity and λ profiles in both methods confirm the intuition from (10): increasing Δt results in increasing the magnitude of the adaptation, as indicated by the step sizes. Note that the TopK algorithm leads to periodically satisfying the sparsity constraint at Δt boundary. This is guaranteed by Proposition 2 (iv), if the no-tie condition is also satisfied. Also note that SGap does not demonstrate the same behavior and it does not guarantee satisfying the sparsity. Overall, selecting small or moderate Δt values appears to be a safer choice in terms of convergence stability and meeting sparsity constraint early in training. The final test accuracy is largely invariant to the choice.

Table 7: Top-1 test accuracy [%] with and without subgradient correction on ResNet-18 and CIFAR-100. The best results per sparsity level and adaptation method are highlighted.

Method	90%	95%	99%
ResNet-18 (Dense, SGD)		$77.7_{\pm 0.2}$	
TopK	$\mathbf{76.5_{\pm 0.1}}$	$\mathbf{75.2_{\pm 0.3}}$	$\mathbf{72.5_{\pm 0.3}}$
TopK + Subgradient correction	$74.9_{\pm 0.4}$	$74.0_{\pm 0.2}$	$70.8_{\pm 0.5}$
SGap	$\mathbf{76.2_{\pm 0.1}}$	$\mathbf{75.3_{\pm 0.1}}$	$\mathbf{73.3_{\pm 0.5}}$
SGap + Subgradient correction	$75.3_{\pm 0.3}$	$73.9_{\pm 0.3}$	$70.9_{\pm 0.2}$

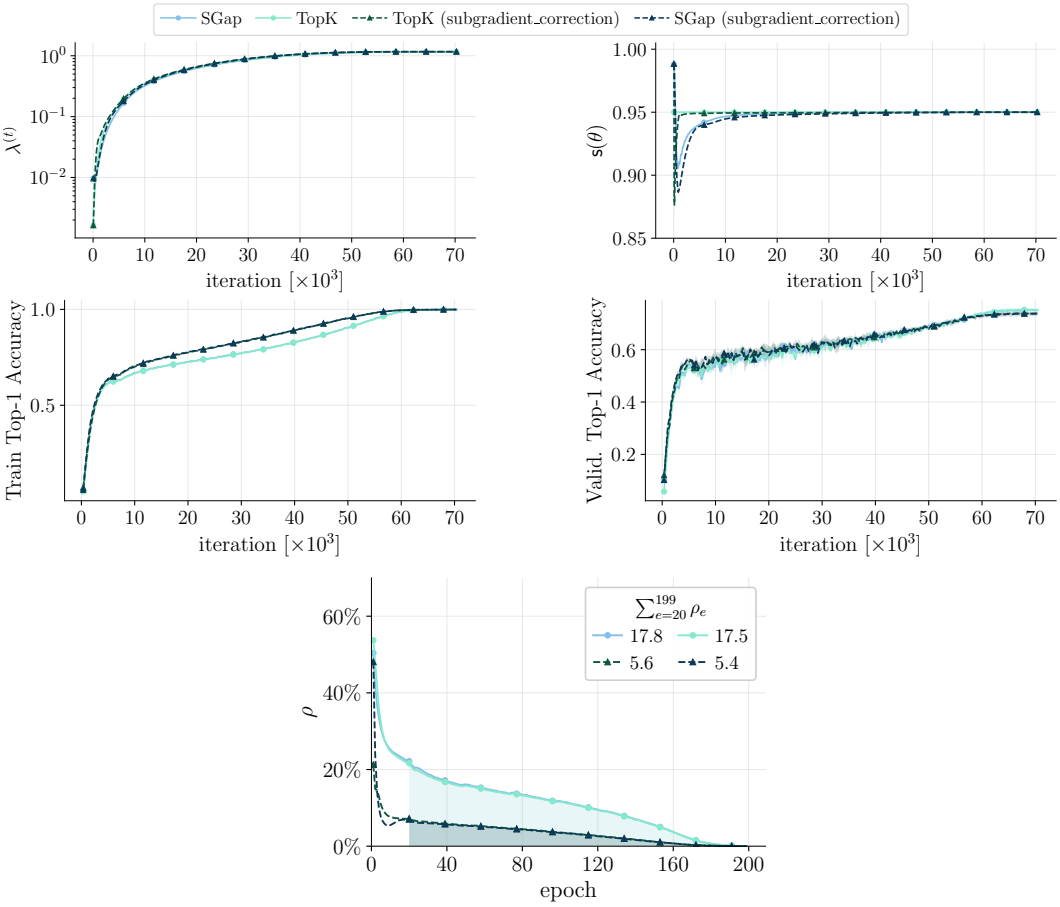


Figure 7: Convergence curves for our baseline algorithms and their subgradient-corrected counterparts measured on ResNet-18 evaluated on CIFAR-100 at $\mathsf{s}^*=95\%$. Note that subgradient-correction leads to dramatically smaller updates to the masks, as indicated by ρ, which is consistent with lower accuracy scores in Table 7.

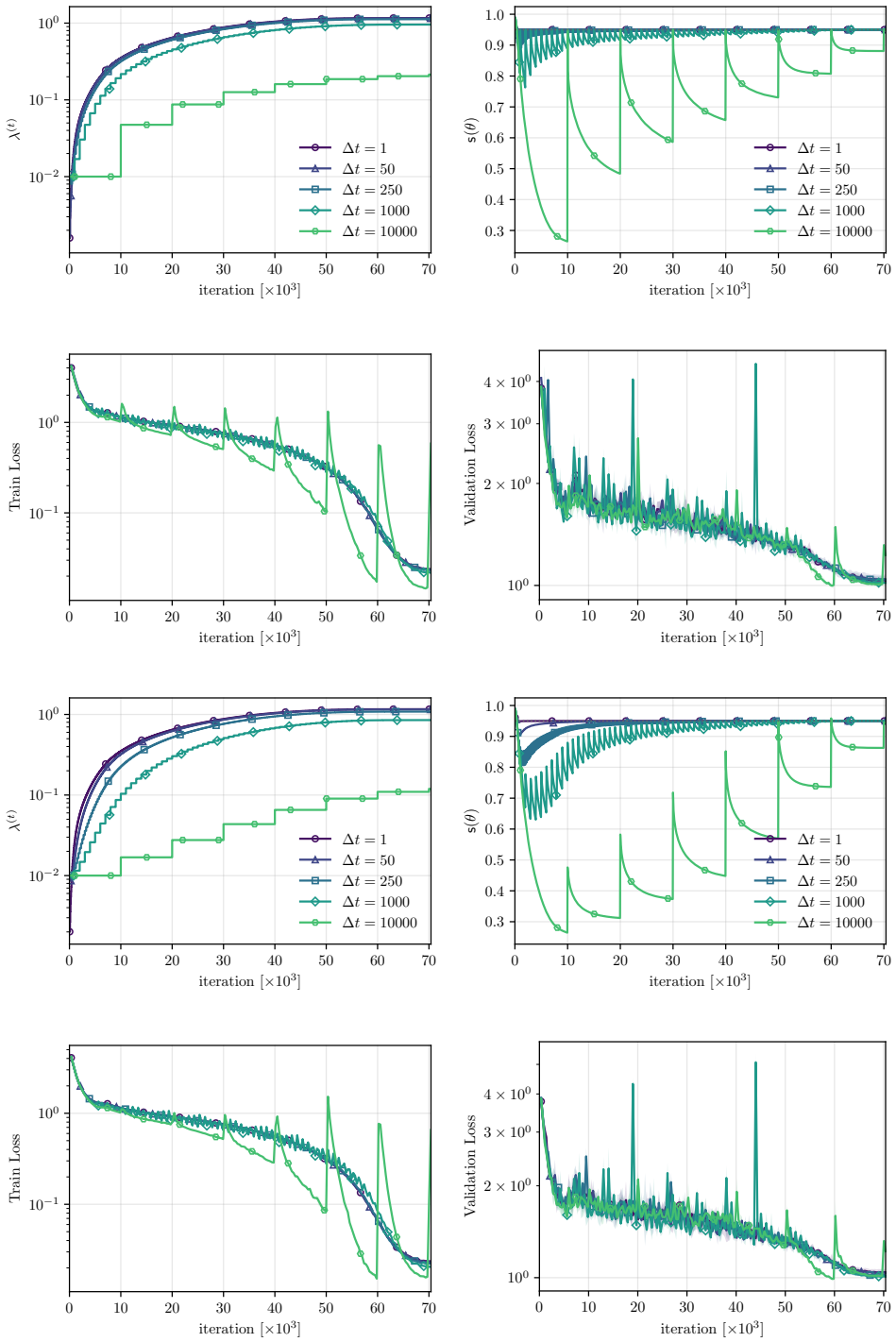


Figure 8: Influence of Δt on TopK (top two rows) and SGap (bottom two rows). For TopK, the sparsity constraint $\mathsf{s}(\theta^{(t)}) = \mathsf{s}^*$ is satisfied with exact equality at Δt boundary. SGap does not necessarily satisfy the sparsity condition each Δt, as in $\Delta t = 10000$. When it does satisfy the condition, it takes more iterations to do so.