Taking the Road Less Scheduled with Adaptive Polyak Steps

Dimitris Oikonomou[1] **Matthew Buchholz**[2] **Yuen-Man Pun**[3]
Robert M. Gower[4] **Nicolas Loizou**[1,4]

[1]Johns Hopkins University [2]University of British Columbia
[3]University of Melbourne [4]Flatiron Institute

{doikono1,nloizou1}@jh.edu, mdbuch@cs.ubc.ca
yuenman.pun@unimelb.edu.au, gowerrobert@gmail.com

Abstract

Schedule-Free SGD, proposed in *The Road Less Scheduled* [Defazio et al., 2024], achieves optimal convergence rates without requiring the training horizon in advance, by replacing learning rate schedules with a principled form of iterate averaging. However, the method still requires tuning a base learning rate whose optimal value depends on unknown problem constants. In this work, we continue down this road by deriving Polyak-type step sizes for Schedule-Free SGD and Adam that compute the learning rate at each iteration from the sampled loss, gradient, and current iterates alone. We first propose an oracle variant that uses per-sample optimal function values and prove an $\mathcal{O}(1/\sqrt{t})$ anytime last-iterate rate for convex Lipschitz objectives. We then remove the oracle requirement with a safeguarded variant that replaces the unknown optimal values with any available lower bound, achieving the same rate up to a neighborhood that vanishes under interpolation. Both step sizes reduce to existing Polyak rules for standard SGD when momentum is set to zero, unifying standard and schedule-free Polyak methods. Numerical experiments on language modeling, including pretraining and distillation, show that the proposed methods match or surpass tuned Schedule-Free baselines while offering greater robustness to hyperparameter choices, and experiments on image classification show that the SGD-based method remains competitive with a tuned baseline without any learning rate tuning.

1 Introduction

We consider the stochastic minimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^d} \; f(\boldsymbol{x}) := \mathbb{E}_{\boldsymbol{\zeta}}[f_{\boldsymbol{\zeta}}(\boldsymbol{x})]\,, \tag{1}$$

where each $f_{\boldsymbol{\zeta}} : \mathbb{R}^d \to \mathbb{R}$ is convex and differentiable. We assume that the set of minimizers $X^* := \arg\min f$ is nonempty and denote $\boldsymbol{x}_\star \in X^*$, $f^\star = f(\boldsymbol{x}_\star)$.

A central practical challenge in training machine learning models is the choice of the learning rate and its schedule. Standard practice relies on learning rate schedules, such as cosine annealing [Loshchilov and Hutter, 2017], linear decay [Hägele et al., 2024], or step-wise reduction [LeCun et al., 1998], that describe how the rate evolves over a pre-specified training horizon T. While highly effective, these schedules share a fundamental limitation: they require the total number of training steps to be fixed in advance, and if training is stopped early or extended beyond T the scheduler is suboptimal. Classical stochastic approximation theory offers a principled alternative through Polyak-Ruppert

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

iterate averaging [Polyak, 1990, Ruppert, 1988], which achieves minimax-optimal convergence rates without knowledge of T. Despite this theoretical appeal, iterate averaging dramatically underperforms in practice [Defazio et al., 2024], creating a long-standing gap between theory and application. Recent work has made significant progress toward closing this gap by developing methods that implicitly apply a schedule through a careful form of iterate averaging, thereby matching or surpassing the practical performance of schedules without requiring the stopping time to be known in advance. Schedule-Free SGD, proposed by Defazio et al. [2024], is a prominent example of this approach.

Schedule-Free SGD maintains three sequences: a *gradient evaluation* sequence $\boldsymbol{y}_t$, a *base* sequence $\boldsymbol{z}_t$ where the gradient step is applied, and an *iterate average* sequence $\boldsymbol{x}_t$ that is returned as output. At each iteration t, the update is

$$\begin{aligned} \boldsymbol{y}_t &= (1-\beta)\boldsymbol{z}_{t-1} + \beta \boldsymbol{x}_t, \\ \boldsymbol{z}_t &= \boldsymbol{z}_{t-1} - \gamma_t \nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \\ \boldsymbol{x}_{t+1} &= (1-c_{t+1})\boldsymbol{x}_t + c_{t+1}\boldsymbol{z}_t, \end{aligned} \qquad \text{(Schedule-Free SGD)}$$

with $c_{t+1} = 1/(t+1)$ and $\boldsymbol{z}_{-1} = \boldsymbol{x}_0$. The momentum parameter $\beta \in [0,1]$ interpolates between Polyak-Ruppert averaging ($\beta = 0$) and Primal averaging ($\beta = 1$); values near $\beta \approx 0.9$ work well in practice [Defazio et al., 2024]. When each $f_{\boldsymbol{\zeta}}$ is G-Lipschitz, Schedule-Free SGD with a constant step size $\gamma_t \equiv \gamma$ achieves the optimal $\mathcal{O}(DG/\sqrt{T})$ rate for any $\beta \in [0,1)$ as shown in Defazio et al. [2024], where $D = \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$. The method has demonstrated strong empirical performance across diverse benchmarks, including winning the MLCommons 2024 AlgoPerf Algorithmic Efficiency Challenge Self-Tuning track[1]. Recently, Brown et al. [2025] established convergence of Schedule-Free SGD in the smooth nonconvex setting for the primal averaging case ($\beta = 1$) with a constant learning rate. The method is also closely related to the AC-SA algorithm of Lan [2012], which achieves optimal rates via a similar averaging scheme. For additional related work, see Appendix A.

Table 1: Summary of convergence guarantees for Schedule-Free SGD under different step size rules. All rates are for convex and Lipschitz objectives and refer to the last iterate $\boldsymbol{x}_t$. "Oracle-free" indicates that the step size does not require knowledge of $f_{\boldsymbol{\zeta}}(\boldsymbol{x}_\star)$. The constant σ^2 is the variance measure defined in (6).

Method	**Step size γ_t**	**Oracle-free?**	**Anytime?**	**Rate**
Schedule-Free SGD [Defazio et al., 2024]	Constant	✓	✗	$\mathcal{O}(1/\sqrt{T})$
Schedule-Free SGD + SF-SPS$_+$ (§2.2)	*Adaptive*	✗	✓	$\mathcal{O}(1/\sqrt{t})$
Schedule-Free SGD + SF-SPS$_{\text{safe}}$ (§2.3)	*Adaptive*	✓	✓	$\mathcal{O}(1/\sqrt{t} + \sigma^2)$

Despite eliminating the dependence on the stopping time T, Schedule-Free SGD still requires tuning the base learning rate γ. The theoretically optimal choice $\gamma = D/(G\sqrt{T})$ depends on the unknown initial distance D, the Lipschitz constant G, and the very quantity T that the method was designed to avoid. In practice, γ is selected by grid search, which can be costly when combined with sweeps over weight decay and momentum.

Instead of tuning γ, here we propose to replace the constant γ with a *Polyak-type* step size [Polyak, 1969], which calculates the learning rate at each iteration from function values rather than from problem constants. The classical Polyak step size and its stochastic extensions [Berrada et al., 2020, Loizou et al., 2021, Garrigos et al., 2023, Oikonomou and Loizou, 2026] have proven effective for SGD, achieving optimal convergence rates with minimal tuning. In this work, we derive Polyak-type step sizes tailored to the Schedule-Free SGD update, making the method fully adaptive. See Table 1 for a summary.

1.1 Main Contributions

Our contributions are summarized as follows.

Polyak step sizes for Schedule-Free SGD. We derive Polyak-type step sizes for Schedule-Free SGD by minimizing an appropriate upper bound on the distance to the solution. The resulting step

[1] Schedule-Free AdamW was the winning entry, see https://mlcommons.org/2024/08/mlc-algoperf-benchmark-competition/

size is closed-form and extends the classical SPS$_+$ rule of Garrigos et al. [2023]. To remove its dependence on the per-sample optimal loss, we introduce a safeguarded variant following the strategy of Oikonomou and Loizou [2026]. In summary, we propose:

- **Oracle (SF-SPS$_+$).** Depends only on the sampled loss $f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)$, the gradient $\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)$, and the current iterates, but requires knowledge of the per-sample optimal loss $f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)$.
- **Safeguarded (SF-SPS$_{\text{safe}}$).** Replaces $f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)$ with any available lower bound $\ell^*_{\boldsymbol{\zeta}_t}$ and prevents exploding steps via a $\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}$ safeguard in the denominator. The resulting step size is fully adaptive: it requires no knowledge of the Lipschitz constant G, the initial distance $\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$, or the stopping time T.

Setting $\beta = 0$ in either variant recovers the corresponding SGD rule of Garrigos et al. [2023], Oikonomou and Loizou [2026], thereby unifying standard and schedule-free Polyak methods within a single framework.

Convergence guarantees. For convex G-Lipschitz objectives, we establish *last-iterate* convergence guarantees for both Polyak step size variants (Section 3). For the oracle step size (SF-SPS$_+$), we prove an $\mathcal{O}(1/\sqrt{t})$ anytime rate that matches the optimum up to constants. For the safeguarded step size (SF-SPS$_{\text{safe}}$), we prove $\mathcal{O}(1/\sqrt{t})$ convergence up to a neighborhood controlled by a variance measure σ^2; this neighborhood vanishes under interpolation. To the best of our knowledge, these are the first convergence guarantees for adaptive Schedule-Free SGD based on Polyak-type step sizes.

Practical variants. We extend the proposed Schedule-Free Polyak step sizes in two practical directions. First, we derive preconditioned Adam-based variants (SF-Adam-SPS$_+$ and SF-Adam-SPS$_{\text{safe}}$) by replacing the Euclidean gradient norm in the denominator with the norm induced by the inverse preconditioner $\boldsymbol{D}_t^{-1}$. These variants recover the SGD rules when $\boldsymbol{D}_t = \boldsymbol{I}$. Second, we introduce an adaptive safeguard that replaces the fixed constant M with an exponential moving average of squared gradient norms, reducing the need to tune this parameter. We evaluate these practical variants alongside the SGD-based methods in our experiments.

Numerical evaluation. We complement our theoretical results with an empirical study across image classification, black-box model distillation, and language model pretraining. On image classification, we run the SGD-based method of Algorithm 1 exactly as analyzed, against Schedule-Free SGD with a tuned learning rate. On model distillation, where the teacher's loss provides a natural approximation of the oracle value, we distill large language models into smaller students using the oracle step size (SF-SPS$_+$). On language modeling, we pretrain a GPT-2 decoder on FineWeb 10B using the Adam-based safeguarded variant. Our results indicate that the proposed Polyak step sizes match or surpass tuned baselines in validation loss on language modeling and remain competitive with a tuned baseline on image classification, while substantially reducing sensitivity to the choice of learning rate, supporting the practical relevance of fully adaptive schedule-free optimization.

2 Polyak Step Sizes for Schedule-Free SGD

This section derives Polyak-type step sizes for Schedule-Free SGD. We begin by recalling two existing Polyak step size rules for SGD: the oracle-based SPS$_+$ of Garrigos et al. [2023] and the safeguarded variant SPS$_{\text{safe}}$ of Oikonomou and Loizou [2026]. We then show how both rules extend naturally to the Schedule-Free setting: first the oracle version (Section 2.2), then the fully adaptive safeguarded version (Section 2.3). The correspondence between the SGD and Schedule-Free rules is summarized in Table 2.

2.1 Background: From SPS$_+$ to SPS$_{\text{safe}}$

A classical strategy for choosing the step size in SGD, given by $\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \gamma_t \nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}^t)$, is to select γ_t that minimizes an upper bound on the quantity $\|\boldsymbol{x}^{t+1} - \boldsymbol{x}_\star\|^2$. When $f_{\boldsymbol{\zeta}_t}$ is convex, expanding the squared distance and bounding the inner product via convexity yields a quadratic in γ_t whose

Table 2: Correspondence between Polyak step sizes for classical SGD and Schedule-Free SGD. Here $\ell^*_{\zeta_t}$ is any lower bound on f_{ζ_t} and $M > 0$ is the safeguard constant. Setting $\beta = 0$ in the Schedule-Free SGD column recovers the corresponding SGD rule, since $\boldsymbol{y}_t = \boldsymbol{z}_{t-1}$.

Setting	**SGD**	**Schedule-Free SGD**
Oracle	$\gamma_t = \dfrac{[f_{\zeta_t}(\boldsymbol{x}^t) - f_{\zeta_t}(\boldsymbol{x}_\star)]_+}{\|\nabla f_{\zeta_t}(\boldsymbol{x}^t)\|^2}$	$\gamma_t = \dfrac{[f_{\zeta_t}(\boldsymbol{y}_t) - f_{\zeta_t}(\boldsymbol{x}_\star) + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle]_+}{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2}$
Safeguarded	$\gamma_t = \dfrac{f_{\zeta_t}(\boldsymbol{x}^t) - \ell^*_{\zeta_t}}{\max\{\|\nabla f_{\zeta_t}(\boldsymbol{x}^t)\|^2, M\}}$	$\gamma_t = \dfrac{\left[f_{\zeta_t}(\boldsymbol{y}_t) - \ell^*_{\zeta_t} + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle\right]_+}{\max\{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2, M\}}$

minimizer over $\gamma_t \geq 0$ is given by the SPS$_+$ rule, as introduced in Garrigos et al. [2023]:

$$\gamma_t = \frac{[f_{\zeta_t}(\boldsymbol{x}^t) - f_{\zeta_t}(\boldsymbol{x}_\star)]_+}{\|\nabla f_{\zeta_t}(\boldsymbol{x}^t)\|^2}, \tag{SPS$_+$}$$

where $[z]_+ = \max\{z, 0\}$ is the ReLU function, and guarantees that $\gamma_t \geq 0$. Under this rule, the iterates are Fejér monotone with respect to $\boldsymbol{x}_\star$, and for convex G-Lipschitz objectives SPS$_+$ achieves the optimal $\mathcal{O}(G\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|/\sqrt{T})$ rate without requiring interpolation [Garrigos et al., 2023]. The price is that SPS$_+$ requires oracle access to $f_{\zeta_t}(\boldsymbol{x}_\star)$, which is rarely available in practice. Earlier variants such as SPS$_{\max}$ of Loizou et al. [2021] sidestep this by using the infima $f^*_{\zeta_t} := \inf f_{\zeta_t}$, but then require the interpolation condition $f_\zeta(\boldsymbol{x}_\star) = \inf_{\boldsymbol{x}} f_\zeta(\boldsymbol{x})$ for all ζ, in order to establish convergence guarantees in the G-Lipschitz setting.

To remove both the oracle dependence and the interpolation requirement, Oikonomou and Loizou [2026] introduced the safeguarded SPS:

$$\gamma_t = \frac{f_{\zeta_t}(\boldsymbol{x}^t) - \ell^*_{\zeta_t}}{\max\{\|\nabla f_{\zeta_t}(\boldsymbol{x}^t)\|^2, M\}}, \tag{SPS$_{\text{safe}}$}$$

where $\ell^*_{\zeta_t} \leq f_{\zeta_t}(\boldsymbol{x}_\star)$ is any available lower bound (typically $\ell^*_{\zeta_t} = 0$ for non-negative losses) and $M > 0$ is a safeguard constant. Two things change relative to SPS$_+$: the unknown $f_{\zeta_t}(\boldsymbol{x}_\star)$ is replaced by the accessible $\ell^*_{\zeta_t}$, and $\max\{\cdot, M\}$ in the denominator prevents the step size from exploding when $\|\nabla f_{\zeta_t}(\boldsymbol{x}^t)\|$ is small. The result is $\mathcal{O}(1/\sqrt{t})$ convergence to a neighborhood of the solution, with the neighborhood vanishing under interpolation [Oikonomou and Loizou, 2026]. We now extend both SPS$_+$ and SPS$_{\text{safe}}$ to Schedule-Free SGD.

2.2 Oracle Polyak Step Size for Schedule-Free SGD

We now carry out the same upper-bound argument for the Schedule-Free $\boldsymbol{z}$ sequence. Recall that the Schedule-Free update sets $\boldsymbol{z}_t = \boldsymbol{z}_{t-1} - \gamma_t \nabla f_{\zeta_t}(\boldsymbol{y}_t)$. Expanding the squared distance to $\boldsymbol{x}_\star$ gives

$$\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2 = \|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2 - 2\gamma_t \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\rangle + \gamma_t^2 \|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2. \tag{2}$$

The key difference from the standard subgradient setting is that the gradient is evaluated at $\boldsymbol{y}_t$, not at $\boldsymbol{z}_{t-1}$. At this point, we cannot minimize the right-hand side in γ_t, because the minima would then depend explicitly on $\boldsymbol{x}_\star$ which appears in the inner product. To remove this explicit dependency on $\boldsymbol{x}_\star$, we will use convexity to bound this inner product.

To apply convexity, we need to decompose $\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star$ in terms of $\boldsymbol{y}_t$. Rearranging the interpolation update $\boldsymbol{y}_t = (1-\beta)\boldsymbol{z}_{t-1} + \beta \boldsymbol{x}_t$ gives

$$\boldsymbol{z}_{t-1} = \boldsymbol{y}_t + \beta(\boldsymbol{z}_{t-1} - \boldsymbol{x}_t) \tag{3}$$

which substituting into the inner product yields

$$\langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\rangle = \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{y}_t - \boldsymbol{x}_\star\rangle + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle.$$

Now convexity of f_{ζ_t} can be applied to the first of the above two terms on the right giving

$$\langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{y}_t - \boldsymbol{x}_\star\rangle \geq f_{\zeta_t}(\boldsymbol{y}_t) - f_{\zeta_t}(\boldsymbol{x}_\star).$$

Substituting back into (2) produces the upper bound

$$\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2 \leq \|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2 + \gamma_t^2 \|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2 - 2\gamma_t \left(f_{\zeta_t}(\boldsymbol{y}_t) - f_{\zeta_t}(\boldsymbol{x}_\star) + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle\right). \tag{4}$$

Minimizing the right-hand side over $\gamma_t \geq 0$ gives the *Schedule-Free Polyak Step*:

$$\boxed{\gamma_t = \frac{[f_{\zeta_t}(\boldsymbol{y}_t) - f_{\zeta_t}(\boldsymbol{x}_\star) + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle]_+}{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2}} \qquad \text{(SF-SPS}_+\text{)}$$

The ReLU function $[\cdot]_+$ guarantees $\gamma_t \geq 0$; it is active only when the momentum correction $\beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle$ is sufficiently negative to overcome $f_{\zeta_t}(\boldsymbol{y}_t) - f_{\zeta_t}(\boldsymbol{x}_\star)$. In the setting where $\beta = 0$, SF-SPS$_+$ becomes SPS$_+$, since in that case $\boldsymbol{y}_t = \boldsymbol{z}_{t-1}$ and the inner product term vanishes.

SF-SPS$_+$ still requires the oracle value $f_{\zeta_t}(\boldsymbol{x}_\star)$; we next introduce a safeguarded variant that replaces it with a lower bound. The complete method for both variants is presented in Algorithm 1.

2.3 Safeguarded Polyak Step Size for Schedule-Free SGD

Following the same strategy that led from SPS$_+$ to SPS$_{\text{safe}}$ in the standard subgradient setting, we apply the same two modifications to remove the oracle dependence of SF-SPS$_+$: (i) the unknown $f_{\zeta_t}(\boldsymbol{x}_\star)$ is replaced by a lower bound $\ell^*_{\zeta_t} \leq f_{\zeta_t}(\boldsymbol{x}_\star)$, and (ii) the denominator $\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2$ is replaced by $\max\{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2, M\}$ for a user-chosen safeguard $M > 0$. This yields the *Safeguarded Schedule-Free Polyak Step*:

$$\boxed{\gamma_t = \frac{\left[f_{\zeta_t}(\boldsymbol{y}_t) - \ell^*_{\zeta_t} + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle\right]_+}{\max\{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2, M\}}} \qquad \text{(SF-SPS}_{\text{safe}}\text{)}$$

The complete method is presented in Algorithm 1.

Algorithm 1 Schedule-Free SGD with SF-SPS$_+$ or SF-SPS$_{\text{safe}}$

Input: $\boldsymbol{z}_{-1} = \boldsymbol{x}_0 \in \mathbb{R}^d$, $\beta \in [0, 1)$, lower bounds $\ell^*_{\zeta_t}$, safeguard $M > 0$
for $t = 0$ **to** $T - 1$ **do**
  $\boldsymbol{y}_t = (1 - \beta)\boldsymbol{z}_{t-1} + \beta \boldsymbol{x}_t$
  Sample ζ_t; compute $\nabla f_{\zeta_t}(\boldsymbol{y}_t)$
  Option I: $\gamma_t = \dfrac{[f_{\zeta_t}(\boldsymbol{y}_t) - f_{\zeta_t}(\boldsymbol{x}_\star) + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle]_+}{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2}$ ▷ SF-SPS$_+$
  or
  Option II: $\gamma_t = \dfrac{\left[f_{\zeta_t}(\boldsymbol{y}_t) - \ell^*_{\zeta_t} + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle\right]_+}{\max\{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2, M\}}$ ▷ SF-SPS$_{\text{safe}}$
  $\boldsymbol{z}_t = \boldsymbol{z}_{t-1} - \gamma_t \nabla f_{\zeta_t}(\boldsymbol{y}_t)$
  $\boldsymbol{x}_{t+1} = (1 - c_{t+1})\boldsymbol{x}_t + c_{t+1}\boldsymbol{z}_t, \quad c_{t+1} = \frac{1}{t+1}$
end for
Return: $\boldsymbol{x}_T$

As in the oracle version, when $\beta > 0$, the inner product term $\beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle$ can be negative, potentially making the numerator negative even though $f_{\zeta_t}(\boldsymbol{y}_t) - \ell^*_{\zeta_t} \geq 0$; the $[\cdot]_+$ operator handles this by clamping the step size to zero. In practice, however, this rarely occurs: in the image classification experiments of Section 5.1, the ReLU is active in at most 2.3% of the steps (Table 5).

When $\beta = 0$, we have $\boldsymbol{y}_t = \boldsymbol{z}_{t-1}$ and the momentum correction vanishes, so SF-SPS$_{\text{safe}}$ reduces to $[f_{\zeta_t}(\boldsymbol{z}_{t-1}) - \ell^*_{\zeta_t}]_+ / \max\{\|\nabla f_{\zeta_t}(\boldsymbol{z}_{t-1})\|^2, M\}$. Since $f_{\zeta_t}(\boldsymbol{z}_{t-1}) - \ell^*_{\zeta_t} \geq 0$, the ReLU function is redundant and we recover exactly SPS$_{\text{safe}}$ applied to the $\boldsymbol{z}$ sequence, with $\boldsymbol{x}_t$ serving as its Polyak-Ruppert average.

Meanwhile, $\max\{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2, M\}$ in the denominator prevents the step size from growing unboundedly when the gradient is small. Unlike the SPS$_{\max}$ approach of Loizou et al. [2021], which

clips the *entire* Polyak step size via $\min\{\gamma_t, \gamma_b\}$ and can reduce the method to a near-constant step size in practice [Oikonomou and Loizou, 2026], our safeguard controls only the denominator and preserves the adaptive numerator throughout training.

Altogether, SF-SPS$_{\text{safe}}$ depends only on quantities available at iteration t, the sampled loss $f_{\zeta_t}(\boldsymbol{y}_t)$, a lower bound $\ell^*_{\zeta_t}$, the gradient $\nabla f_{\zeta_t}(\boldsymbol{y}_t)$, and the current iterates, and requires no knowledge of the Lipschitz constant G, the initial distance $\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$, or the stopping time T.

3 Convergence Analysis

This section presents convergence guarantees for Schedule-Free SGD equipped with the Polyak step sizes derived in Section 2. We first analyze the oracle step size SF-SPS$_+$, which yields a clean $\mathcal{O}(1/\sqrt{t})$ last-iterate rate but requires knowledge of $f_{\zeta_t}(\boldsymbol{x}_\star)$ (Section 3.1). We then turn to the safeguarded step size SF-SPS$_{\text{safe}}$, which removes this oracle dependence at the cost of convergence to a neighborhood (Section 3.2). Complete proofs are deferred to the appendix.

3.1 Convergence with Oracle Step Size

Theorem 3.1. Consider the iterates of Schedule-Free SGD with the oracle step size SF-SPS$_+$ and $c_t = 1/(t+1)$. Let $\beta \in [0, 1)$ and let $f_\zeta : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable for every ζ. Define

$$B := \{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x} - \boldsymbol{x}_\star\| \le \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|\}$$
$$G^2 := \max_{\boldsymbol{x} \in B} \mathbb{E}\left[\|\nabla f_\zeta(\boldsymbol{x})\|^2\right].$$

With the initialization $\boldsymbol{z}_{-1} = \boldsymbol{x}_0$, the last iterate satisfies, for all $t \ge 1$,

$$\mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right] \le \frac{G\,\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|}{\sqrt{t+1}}. \tag{5}$$

The bound (5) has two notable features. First, it is an *anytime* result: it achieves the $\mathcal{O}(1/\sqrt{t})$ rate at every iteration t, without requiring the stopping time T to be specified in advance. This is in contrast to the constant-step analysis of Defazio et al. [2024], where the optimal choice $\gamma = D/(G\sqrt{T})$ depends on T. Second, the constant G is defined locally over the ball B rather than globally. The proof establishes that the iterates $\boldsymbol{z}_t$ are Fejér monotone with respect to $\boldsymbol{x}_\star$, i.e., $\|\boldsymbol{z}_t - \boldsymbol{x}_\star\| \le \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$ for all t. Since $\boldsymbol{x}_t$ is a convex combination of the $\boldsymbol{z}$ iterates and $\boldsymbol{y}_t$ is a convex combination of $\boldsymbol{z}_{t-1}$ and $\boldsymbol{x}_t$, all three sequences remain in B, and consequently the gradients evaluated during training satisfy $\mathbb{E}\left[\|\nabla f_\zeta(\boldsymbol{y}_t)\|^2\right] \le G^2$.

The rate (5) matches, up to a constant factor, the optimal rate $\mathcal{O}(DG/\sqrt{T})$ for convex Lipschitz optimization [Orabona, 2019]. The practical limitation is that SF-SPS$_+$ requires oracle access to $f_{\zeta_t}(\boldsymbol{x}_\star)$. While this quantity is available in some settings, for instance, under interpolation where $f_\zeta(\boldsymbol{x}_\star) = \inf_{\boldsymbol{x}} f_\zeta(\boldsymbol{x})$, or in black-box model distillation, it is unavailable in general. This motivates the safeguarded analysis below, which replaces the oracle value with a lower bound at the cost of convergence to a neighborhood.

3.2 Convergence with Safeguarded Step Size

The gap between the lower bound $\ell^*_{\zeta_t}$ and the true optimal value $f_{\zeta_t}(\boldsymbol{x}_\star)$ introduces a noise term that prevents exact convergence. Following the stochastic Polyak step size literature [Loizou et al., 2021, Oikonomou and Loizou, 2026], we quantify this gap via the variance measure

$$\sigma^2 := \left(\mathbb{E}\left[\left(f_{\zeta_t}(\boldsymbol{x}_\star) - \ell^*_{\zeta_t}\right)^2\right]\right)^{1/2}. \tag{6}$$

Since each f_ζ is lower bounded, $\sigma^2 < \infty$. When the problem is *interpolated*, i.e., there exists $\boldsymbol{x}_\star$ such that $f_\zeta(\boldsymbol{x}_\star) = \inf_{\boldsymbol{x}} f_\zeta(\boldsymbol{x})$ for all ζ, choosing $\ell^*_\zeta = \inf_{\boldsymbol{x}} f_\zeta(\boldsymbol{x})$ gives $\sigma^2 = 0$. The interpolation condition is satisfied in many over-parameterized settings [Ma et al., 2018, Zhang et al., 2021].

Theorem 3.2. Consider the iterates of Schedule-Free SGD with the safeguarded step size SF-SPS$_{\text{safe}}$ and $c_t = 1/(t+1)$. Let $f_{\boldsymbol{\zeta}} : \mathbb{R}^d \to \mathbb{R}$ be convex, differentiable, and G-Lipschitz for every $\boldsymbol{\zeta}$. With the initialization $\boldsymbol{z}_{-1} = \boldsymbol{x}_0$, the last iterate satisfies, for all $t \geq 1$,

$$\mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right] + \frac{\beta}{1-\beta}\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}\left[B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right]$$
$$\leq \frac{\sqrt{\max\{G^2, M\}}\,\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|}{\sqrt{t+1}} + \sqrt{\frac{\max\{G^2, M\}}{M}}\,\sigma^2, \tag{7}$$

where $B_f(\boldsymbol{x}, \boldsymbol{y}) := f(\boldsymbol{x}) - f(\boldsymbol{y}) - \langle \nabla f(\boldsymbol{y}), \boldsymbol{x} - \boldsymbol{y}\rangle \geq 0$ is the Bregman divergence of f.

Theorem 3.2 provides the same $\mathcal{O}(1/\sqrt{t})$ leading rate as Theorem 3.1, up to a neighborhood whose size is controlled by σ^2. The critical difference is that SF-SPS$_{\text{safe}}$ requires only a lower bound $\ell^*_{\boldsymbol{\zeta}_t}$ rather than the oracle value $f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)$, making the method fully adaptive. There is also a difference in assumptions: Theorem 3.1 requires G-Lipschitz continuity only locally on the ball B, while Theorem 3.2 requires it globally. This is because the safeguarded step size does not guarantee Fejér monotonicity of the $\boldsymbol{z}$ iterates, so the iterates cannot be confined to B a priori.

Furthermore, the left-hand side of (7) contains, in addition to the suboptimality gap, a non-negative Bregman divergence term weighted by $\beta/(1-\beta)$. Since the bound holds for the sum of two non-negative quantities, it implies in particular that $\mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right]$ is bounded by the right-hand side. The Bregman term measures how far the evaluation sequence $\boldsymbol{x}_k$ departs from the gradient-location sequence $\boldsymbol{y}_k$, and its presence on the left-hand side suggests that these sequences stay close throughout training. When $\beta = 0$, the coefficient $\beta/(1-\beta)$ vanishes, the Bregman term drops out, and Theorem 3.2 reduces to the SPS$_{\text{safe}}$ guarantee of Oikonomou and Loizou [2026].

4 Practical Variants

The step size rules introduced in Section 2 and analyzed in Section 3 are derived for the SGD-based Schedule-Free update. To make the method more broadly applicable in practice, we now discuss two extensions used in our experiments: a preconditioned Adam variant and an adaptive safeguard.

4.1 Preconditioned Extension: Adam

Adaptive gradient methods such as Adam [Kingma and Ba, 2015] and its decoupled weight-decay variant AdamW [Loshchilov and Hutter, 2019] have become the dominant optimizers for training deep networks. This line of adaptive preconditioning, building on AdaGrad [Duchi et al., 2011] and RMSProp [Tieleman and Hinton, 2012], is especially critical for problems with heterogeneous curvature such as transformer training. The Schedule-Free framework has already been shown to combine effectively with this preconditioner, achieving state-of-the-art results across a wide range of tasks [Defazio et al., 2024], which motivates extending our step size rules to this setting.

Let $\boldsymbol{D}_t \succ 0$ be a symmetric positive definite preconditioner and let $\|\boldsymbol{v}\|^2_{\boldsymbol{D}_t^{-1}} := \boldsymbol{v}^\top \boldsymbol{D}_t^{-1}\boldsymbol{v}$. Replacing the gradient step with $\boldsymbol{z}_t = \boldsymbol{z}_{t-1} - \gamma_t \boldsymbol{D}_t^{-1}\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)$ and expanding $\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2_{\boldsymbol{D}_t}$ via the same argument as in Section 2.2 yields an upper bound identical to (4) with $\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2$ replaced by $\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2_{\boldsymbol{D}_t^{-1}}$. Minimizing over $\gamma_t \geq 0$, and applying the oracle-to-safeguarded modifications of Section 2.3, gives

$$\gamma_t = \frac{\left[f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t) - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) + \beta\,\langle \nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle\right]_+}{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2_{\boldsymbol{D}_t^{-1}}}, \tag{SF-Adam-SPS$_+$}$$

$$\gamma_t = \frac{\left[f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t) - \ell^*_{\boldsymbol{\zeta}_t} + \beta\,\langle \nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle\right]_+}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2_{\boldsymbol{D}_t^{-1}}, M\}}. \tag{SF-Adam-SPS$_{\text{safe}}$}$$

Setting $\boldsymbol{D}_t = \boldsymbol{I}$ recovers SF-SPS$_+$ and SF-SPS$_{\text{safe}}$, respectively. The complete procedure, instantiated with the standard Adam diagonal preconditioner, is given in Algorithm 2.

Remark 4.1 (Practical details)**.** Since $\boldsymbol{y}_t = (1-\beta)\boldsymbol{z}_{t-1} + \beta\boldsymbol{x}_t$ implies $\beta(\boldsymbol{z}_{t-1} - \boldsymbol{x}_t) = \boldsymbol{z}_{t-1} - \boldsymbol{y}_t$, the momentum correction appearing in all four step sizes (SF-SPS$_+$, SF-SPS$_{\text{safe}}$, SF-Adam-SPS$_+$, SF-Adam-SPS$_{\text{safe}}$) satisfies

$$\beta \langle \nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle = \langle \nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{y}_t \rangle ,$$

so the numerator can be computed directly from $\boldsymbol{z}_{t-1}$ and $\boldsymbol{y}_t$, avoiding the need to store $\boldsymbol{x}_t$.

4.2 Adaptive Safeguard via Exponential Moving Average

The safeguarded step sizes SF-SPS$_{\text{safe}}$ and SF-Adam-SPS$_{\text{safe}}$ require choosing the constant $M > 0$. While any positive value guarantees bounded step sizes, the choice affects practical performance: too large and the safeguard dominates; too small and it fails to prevent occasional large steps. Prior work on SPS$_{\text{safe}}$ for standard SGD observed similar sensitivity and proposed replacing the fixed M with an exponential moving average of past squared gradient norms [Oikonomou and Loizou, 2026]. We adopt the same strategy for the Schedule-Free setting. Concretely, the safeguard is updated as

$$M_t = \beta_M \, M_{t-1} + (1 - \beta_M) \, \|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2 , \tag{8}$$

with $M_0 = \|\nabla f_{\boldsymbol{\zeta}_0}(\boldsymbol{y}_0)\|^2$, and the fixed M in SF-SPS$_{\text{safe}}$ or SF-Adam-SPS$_{\text{safe}}$ is replaced by M_t. The smoothing parameter $\beta_M \in [0, 1)$ controls the memory of the estimate; we recommend $\beta_M = 0.99$ as a robust default. This adaptive safeguard tracks the scale of the gradients throughout training without introducing additional hyperparameters that require careful tuning.

5 Experiments

Our experiments evaluate the proposed Polyak step sizes across three settings, each targeting a different aspect of the method: image classification (Section 5.1), where we run the SGD-based method of Algorithm 1 exactly as analyzed in Section 3; black-box model distillation (Section 5.2), where the teacher's loss approximates the oracle value and allows us to test SF-SPS$_+$; and language model pretraining (Section 5.3), where we evaluate SF-Adam-SPS$_{\text{safe}}$ at scale. Our main finding is that the proposed step sizes match or exceed tuned baselines in validation loss on language modeling, remain competitive with a tuned baseline on image classification, and exhibit substantially greater robustness to hyperparameter choices throughout. We provide the code for our experiments at `https://github.com/buchholzmd/GPT-opt`.

5.1 Image Classification

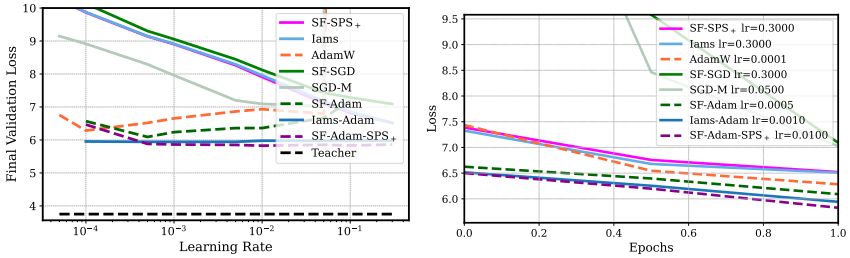
We begin with the method analyzed in Section 3, before turning to the Adam-based variants of Section 4 in the remaining experiments. We run Algorithm 1 with SF-SPS$_{\text{safe}}$ verbatim, including the harmonic averaging weights $c_{t+1} = 1/(t+1)$, so that there is no gap between the analyzed method and the evaluated one. Since no preconditioner is involved, any robustness we observe is attributable to the Polyak step size itself.

We train ResNet-20 and ResNet-32 [He et al., 2016] on CIFAR-10 and CIFAR-100 [Krizhevsky, 2009] for 200 epochs with batch size 128, momentum $\beta = 0.9$, and no weight decay. The baseline is Schedule-Free SGD with a constant learning rate swept over the six values $\gamma \in \{0.25, 0.5, 1, 2, 5, 10\}$, of which we report the best and the worst member. SF-SPS$_{\text{safe}}$ uses the defaults used throughout this section, namely $\ell^*_{\boldsymbol{\zeta}_t} = 0$ and either the EMA safeguard of Section 4.2 with $\beta_M = 0.99$ or a fixed M. Following Defazio et al. [2024], we evaluate the averaged iterate $\boldsymbol{x}_t$ after every epoch and report the best test accuracy; the complete protocol is given in Appendix C.2.

The results are presented in Table 3. We observe that, with no tuning, SF-SPS$_{\text{safe}}$ lands within three points of the sweep-best test accuracy in every setting. Moreover, the best learning rate of the baseline moves across settings ($\gamma = 1$ for ResNet-20 and $\gamma = 0.5$ for ResNet-32), whereas the defaults of SF-SPS$_{\text{safe}}$ are the same throughout. Regarding the safeguard, the test accuracy varies by at most 1.5 points across $M \in \{1, 2, 5, 10\}$ and the EMA safeguard in each setting, whereas for $M \in \{20, 50\}$ the safeguard binds in nearly every step and the accuracy drops by 1 to 5 points; the EMA safeguard binds in roughly half of the steps and removes the need to choose M altogether. We report the final training losses and the safeguard activation statistics in Appendix C.2.

Table 3: Test accuracy (%) on CIFAR-10/100 after 200 epochs. Schedule-Free SGD uses a constant learning rate swept over six values (best and worst member reported, learning rate in parentheses). SF-SPS$_{\text{safe}}$ requires no sweep and uses $\ell^*_{\zeta_t} = 0$ with the EMA safeguard ($\beta_M = 0.99$) or a fixed M.

	Schedule-Free SGD (LR sweep)		SF-SPS$_{\text{safe}}$ (no sweep)						
Setting	best (LR)	worst (LR)	EMA M	M=1	M=2	M=5	M=10	M=20	M=50
CIFAR-10 / ResNet-20	88.70 (1)	82.02 (10)	86.26	86.50	86.53	86.06	85.08	84.04	81.21
CIFAR-10 / ResNet-32	89.35 (0.5)	81.40 (10)	85.63	86.07	86.06	86.11	86.05	84.77	81.67
CIFAR-100 / ResNet-20	61.73 (1)	55.74 (10)	59.84	59.48	60.44	60.60	59.21	59.18	55.66
CIFAR-100 / ResNet-32	60.91 (0.5)	50.35 (10)	58.75	59.52	59.96	58.69	59.82	57.48	56.47



(a) `tiny_shakespeare`: LR sensitivity (b) `tiny_shakespeare`: validation loss

Figure 1: Black-box model distillation results. Distilling `gpt2-medium` into a 67.7M-parameter student on `tiny_shakespeare`. Left: final validation loss as a function of the base learning rate. Right: validation loss curves for the best learning rate of each method.

5.2 Model Distillation

The oracle step size SF-SPS$_+$ requires access to the per-sample optimal value $f_{\zeta_t}(\boldsymbol{x}_\star)$, which is unavailable in general. Black-box model distillation offers a natural setting where this quantity can be approximated. Let f^t_ζ and $f^s_\zeta(\boldsymbol{x})$ denote the teacher's and student's loss on batch ζ, respectively. When the teacher has been trained to near-optimality on a large corpus and the student has sufficient capacity to match it, the teacher loss satisfies $f^t_\zeta \approx f^s_\zeta(\boldsymbol{x}_\star)$, where $\boldsymbol{x}_\star$ denotes the optimal student parameters.

We test the Adam-based oracle variant SF-Adam-SPS$_+$ (see Algorithm 2 for pseudocode). Our setup follows Gower et al. [2025]: we compare against SGD with momentum, Adam [Kingma and Ba, 2015], (Adam-)ScheduleFree [Defazio et al., 2024], and Iams(-Adam) [Gower et al., 2025]. For every baseline we sweep the base learning rate. For SF-Adam-SPS$_+$, the step size is computed adaptively via $\gamma_t = \min\{\gamma_{\max}, \tau_t\}$, where τ_t is the oracle Polyak step and $\gamma_{\max} > 0$ is a maximum step size that prevents excessively large updates when the teacher-loss approximation is imprecise. This cap is a standard safeguard in stochastic Polyak methods [Loizou et al., 2021]; we sweep $\gamma_{\max}$ in the same manner as the base learning rate for the baselines.

Distilling `tiny_shakespeare`. The teacher model employed was `gpt2-medium` (345 million parameters), a pre-trained transformer model from the Hugging Face library [Radford et al., 2019]. We used a student model with 67.7 million parameters, see Table 6 in Appendix C.3 for details. The results in Figure 1b show that our SF-Adam-SPS$_+$ achieves the best loss for a tuned learning rate $\gamma_{\max}$, and is as robust as the Iams-Adam method to the choice of learning rate.

5.3 Language Modeling

We test SF-Adam-SPS$_{\text{safe}}$ on pretraining GPT-2 [Radford et al., 2019] style decoders on the FineWeb 10B dataset [Penedo et al., 2024], matching the architecture, batch size, number of training iterations, and schedules of Defazio et al. [2024]. Following Defazio et al. [2024], we replace the theoretical averaging weights $c_{t+1} = 1/(t+1)$ with the practical heuristic $c_{t+1} = \gamma_t^2 / \sum_{i=1}^t \gamma_i^2$ and use a warmup-stable schedule. We compare against AdamW and Schedule-Free AdamW, sweeping the base learning rate for both baselines over six values (see Tables 7 to 9 in Appendix C for full details). For SF-Adam-SPS$_{\text{safe}}$, the step size is computed adaptively and requires no learning rate sweep and

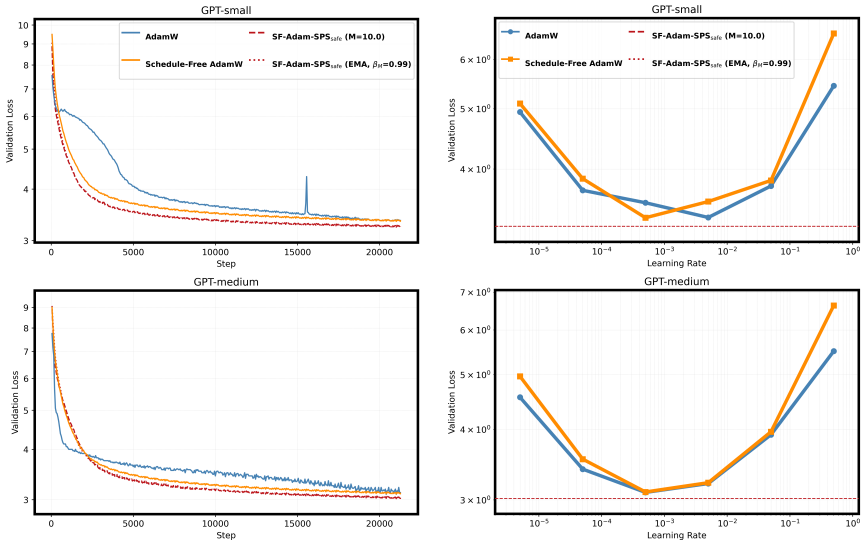


Figure 2: Validation loss (*left*) and LR sensitivity (*right*) for pretraining GPT2-small (124M parameters, *top*) and GPT2-medium (355M parameters, *bottom*) on the `fineweb10B` data set.

introduces no additional per-step overhead, since the loss is already computed during the forward pass. We evaluate two safeguard strategies: a fixed constant $M = 10$ and the EMA-based adaptive safeguard from Section 4.2 with $\beta_M = 0.99$.

Figure 2 shows results for GPT-2 (small). In terms of training loss, SF-Adam-SPS$_{\text{safe}}$ matches Schedule-Free AdamW across both M settings, with both outperforming AdamW. In terms of validation loss, SF-Adam-SPS$_{\text{safe}}$ achieves the best performance of all methods regardless of M setting, suggesting better generalization. The LR sensitivity plot confirms that SF-Adam-SPS$_{\text{safe}}$ requires no learning rate tuning, remaining flat across the full sweep while AdamW and Schedule-Free AdamW degrade sharply outside a narrow optimal range. We plot the adaptive Polyak step-sizes in Figure 3a and plot the safeguard M against preconditioned gradient norms in Figure 4.

Computational cost. Beyond matching or exceeding tuned baselines in validation loss, the adaptive step size yields a substantial reduction in total compute. Finding the best learning rate for AdamW and Schedule-Free AdamW required sweeping over six values, consuming approximately 220 and 195 GPU-hours respectively on 4×H100s. By contrast, SF-Adam-SPS$_{\text{safe}}$ computes its step size adaptively and requires no sweep: a single run completes in approximately 21 GPU-hours on 4×A100s: this would be faster on equivalent H100 hardware, since the H100 offers roughly 2–3× higher throughput for mixed-precision training. This makes the method particularly attractive in large-scale settings where learning rate tuning constitutes a significant fraction of the total training budget.

6 Conclusion

In this work, we derive Polyak-type step sizes for Schedule-Free SGD, yielding adaptive learning rates that require no knowledge of the Lipschitz constant, the initial distance to the optimum, or the stopping time. We provide last-iterate convergence guarantees for both an oracle and a safeguarded variant, achieving an $\mathcal{O}(1/\sqrt{t})$ rate for the oracle variant and the same rate up to a neighborhood for the safeguarded variant. These results offer a principled route to fully adaptive schedule-free optimization, reducing the tuning burden to a single safeguard constant. Future directions include extending the convergence analysis to smooth or strongly convex objectives to obtain faster rates, establishing guarantees for the Adam-based extension, and investigating tighter variance measures that could shrink or eliminate the convergence neighborhood beyond the interpolation regime.

Acknowledgments and Disclosure of Funding

The work of Dimitris Oikonomou and Nicolas Loizou was supported by NSF CAREER Award 2542902 and AFOSR YIP Award FA9550-26-1-B353.

References

Farshed Abdukhakimov, Cuong Anh Pham, Samuel Horvath, Martin Takac, and Slavomir Hanzely. Polyak stepsize: Estimating optimal functional values without parameters or prior knowledge. *arXiv preprint arXiv:2508.17288*, 2025.

Leonard Berrada, Andrew Zisserman, and M Pawan Kumar. Training neural networks for and by interpolation. In *ICML*, 2020.

Connor Brown, Ahmed Khaled, and Chi Jin. Analysis of schedule-free nonconvex optimization. In *NeurIPS*, 2025.

Tri Dao, Daniel Y Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *NeurIPS*, 2022.

Aaron Defazio. ScheduleFree+: Scaling learning-rate-free & schedule-free learning to large language models. *arXiv preprint arXiv:2605.19095*, 2026.

Aaron Defazio, Xingyu Yang, Ahmed Khaled, Konstantin Mishchenko, Harsh Mehta, and Ashok Cutkosky. The road less scheduled. In *NeurIPS*, 2024.

Ryan D'Orazio, Nicolas Loizou, Issam H Laradji, and Ioannis Mitliagkas. Stochastic mirror descent: Convergence analysis and adaptive variants via the mirror stochastic Polyak stepsize. *Transactions on Machine Learning Research*, 2023.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

Guillaume Garrigos, Robert M Gower, and Fabian Schaipp. Function value learning: Adaptive learning rates based on the Polyak stepsize and function splitting in ERM. *arXiv preprint arXiv:2307.14528*, 2023.

Robert Gower, Othmane Sebbouh, and Nicolas Loizou. SGD for structured nonconvex functions: Learning rates, minibatching and interpolation. In *AISTATS*, 2021.

Robert M Gower, Mathieu Blondel, Nidham Gazagnadou, and Fabian Pedregosa. Cutting some slack for SGD with adaptive Polyak stepsizes. *arXiv preprint arXiv:2202.12328*, 2022.

Robert M Gower, Guillaume Garrigos, Nicolas Loizou, Dimitris Oikonomou, Konstantin Mishchenko, and Fabian Schaipp. Analysis of an idealized stochastic Polyak method and its application to black-box model distillation. *arXiv preprint arXiv:2504.01898*, 2025.

Alex Hägele, Elie Bakouch, Atli Kosson, Leandro Von Werra, Martin Jaggi, et al. Scaling laws and compute-optimal training beyond fixed training durations. In *NeurIPS*, 2024.

Elad Hazan and Sham Kakade. Revisiting the Polyak step size. *arXiv preprint arXiv:1905.00313*, 2019.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

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

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Guanghui Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1):365–397, 2012.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Nicolas Loizou, Sharan Vaswani, Issam Hadj Laradji, and Simon Lacoste-Julien. Stochastic Polyak step-size for SGD: An adaptive learning rate for fast convergence. In *AISTATS*, 2021.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *ICLR*, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning. In *ICML*, pages 3325–3334, 2018.

Sohom Mukherjee, Nicolas Loizou, and Sebastian U Stich. Locally adaptive federated learning. *Transactions on Machine Learning Research*, 2024.

Dimitris Oikonomou and Nicolas Loizou. Stochastic polyak step-sizes and momentum: Convergence guarantees and practical performance. In *ICLR*, 2025.

Dimitris Oikonomou and Nicolas Loizou. Safeguarded stochastic polyak step-sizes for non-smooth optimization: Robust performance without small (sub)gradients. In *ICML*, 2026.

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

Francesco Orabona and Ryan D'Orazio. New perspectives on the Polyak stepsize: Surrogate functions and negative results. In *NeurIPS*, 2025.

Antonio Orvieto and Lin Xiao. An adaptive stochastic gradient method with non-negative Gauss-Newton stepsizes. *arXiv preprint arXiv:2407.04358*, 2024.

Antonio Orvieto, Simon Lacoste-Julien, and Nicolas Loizou. Dynamics of SGD with stochastic Polyak stepsizes: Truly adaptive variants and convergence to exact solution. In *NeurIPS*, 2022.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, et al. The FineWeb datasets: Decanting the web for the finest text data at scale. In *NeurIPS*, 2024.

Boris T Polyak. New stochastic approximation type procedures. *Avtomatika i Telemekhanika*, (7): 98–107, 1990.

Boris Teodorovich Polyak. Minimization of unsmooth functionals. *USSR Computational Mathematics and Mathematical Physics*, 9(3):14–29, 1969.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI Blog*, 2019.

David Ruppert. Efficient estimations from a slowly convergent Robbins-Monro process. Technical report, Cornell University Operations Research and Industrial Engineering, 1988.

Fabian Schaipp, Robert M. Gower, and Michael Ulbrich. A stochastic proximal Polyak step size. *Transactions on Machine Learning Research*, 2023.

Fabian Schaipp, Ruben Ohana, Michael Eickenberg, Aaron Defazio, and Robert M Gower. MoMo: Momentum models for adaptive learning rates. In *ICML*, 2024.

Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5 – RMSProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, University of Toronto, 2012.

Xiaoyu Wang, Mikael Johansson, and Tong Zhang. Generalized Polyak step size for first order optimization with momentum. In *ICML*, 2023.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Supplementary Material

The Supplementary Material begins with a broader discussion of related work on Polyak-type step sizes (Appendix A), followed by the auxiliary lemmas and complete proofs of our convergence guarantees (Appendix B), and concludes with additional experimental results (Appendix C).

Contents

1 Introduction **1**

1.1 Main Contributions 2

2 Polyak Step Sizes for Schedule-Free SGD **3**

2.1 Background: From SPS_+ to SPS_{safe} 3

2.2 Oracle Polyak Step Size for Schedule-Free SGD 4

2.3 Safeguarded Polyak Step Size for Schedule-Free SGD 5

3 Convergence Analysis **6**

3.1 Convergence with Oracle Step Size 6

3.2 Convergence with Safeguarded Step Size 6

4 Practical Variants **7**

4.1 Preconditioned Extension: Adam 7

4.2 Adaptive Safeguard via Exponential Moving Average 8

5 Experiments **8**

5.1 Image Classification 8

5.2 Model Distillation 9

5.3 Language Modeling 9

6 Conclusion **10**

A Related Work on Polyak Step Sizes **14**

B Proofs of Convergence Guarantees **15**

B.1 Auxiliary Lemmas 15

B.2 Proof of Theorem 3.1 (Oracle Step Size) 18

B.3 Proof of Theorem 3.2 (Safeguarded Step Size) 18

C Additional Experimental Results **21**

C.1 Pseudocodes 21

C.2 Image Classification with SF-SPS_{safe} 21

C.3 Black-Box Distillation details 22

C.4 Pretraining GPT-2 Decoder Experiment Details 23

A Related Work on Polyak Step Sizes

The Polyak step size [Polyak, 1969] computes the learning rate at each iteration from the current function value, the optimal value $f^\star$, and the gradient norm, removing the need to specify a base learning rate. While originally analyzed for convex Lipschitz objectives, Hazan and Kakade [2019] broadened its theoretical foundations to cover smooth and strongly convex settings as well, and proposed a variant that estimates $f^\star$ online rather than requiring it as input. Extending these ideas to stochastic optimization, Loizou et al. [2021] proposed $\text{SPS}_{\max}$ and established convergence in two settings: convex smooth objectives and, separately, convex Lipschitz objectives, with the latter requiring the interpolation condition. Garrigos et al. [2023] relaxed this by introducing the SPS_+ rule, which achieves the optimal rate in the Lipschitz setting without interpolation, at the cost of requiring per-sample optimal function values; they also proposed Function Value Learning to estimate these quantities in practice. To remove this dependence as well, Oikonomou and Loizou [2026] introduced a denominator safeguard that replaces the per-sample optimal values with any available lower bound, yielding convergence guarantees for non-smooth objectives under minimal assumptions.

Several works have refined and broadened SPS-type methods. Orvieto et al. [2022] identified biases introduced by stochastic Polyak steps outside interpolation and proposed decreasing-step variants that converge exactly. Orabona and D'Orazio [2025] gave a complementary surrogate-loss perspective, interpreting the Polyak step size and several stochastic variants as gradient descent on non-negative surrogate functions, and also established negative results showing that some Polyak variants may converge only to a neighborhood outside favorable regimes such as interpolation. Gower et al. [2022] unified SPS-type rules through a cutting-plane perspective, introducing slack variants for non-interpolated settings. For composite objectives, Schaipp et al. [2023] developed a proximal Polyak step size that handles non-smooth regularizers. D'Orazio et al. [2023] generalized the framework to mirror descent in non-Euclidean geometries and Gower et al. [2021] established Polyak step size guarantees for the broader class of quasar-convex functions. In the distributed setting, Mukherjee et al. [2024] employed locally adaptive Polyak-type step sizes within a federated learning framework. Another line of work removes the need to know the optimal function value by estimating it online; for example, Abdukhakimov et al. [2025] proposed a parameter-free Polyak step size for deterministic gradient descent that maintains auxiliary iterates to estimate $f^\star$ during the run.

Closer to our setting are Polyak-type rules for momentum methods. Schaipp et al. [2024] proposed MoMo, which builds momentum-based models of sampled losses to produce adaptive step sizes compatible with heavy-ball updates. Oikonomou and Loizou [2025] designed Polyak step sizes for stochastic heavy-ball with guarantees both to neighborhoods and to exact minimizers via decreasing variants. Wang et al. [2023] explored adaptive Polyak-type rules in momentum settings. Orvieto and Xiao [2024] connected Polyak-like adaptivity to second-order modeling by exploiting non-negative losses, writing them as squares, and applying stochastic Gauss-Newton or Levenberg-Marquardt steps that automatically warm up and decay with the loss landscape. Gower et al. [2025] analyzed an idealized stochastic Polyak method for momentum and applied it to black-box model distillation, which directly motivates our experimental setup.

The step size proposed in this work has already been adopted at scale. Building on SF-SPS_+, the recent ScheduleFree+ of Defazio [2026] scales learning-rate-free Schedule-Free training to large language models with up to 2B parameters trained on up to one trillion tokens. Its step size keeps the numerator of SF-SPS_+, the sampled loss plus the momentum correction, but replaces the squared Euclidean norm in the denominator with the ℓ_1 norm of the gradient scaled by $\sqrt{\pi/2}$, optionally averaged over past steps, and combines the resulting step with learning rate warmup, an Adam-type preconditioner with decoupled weight decay, and an increasing outer momentum. Defazio [2026] reports that this combination outperforms tuned warmup-stable-decay and linear decay schedules, with an advantage that grows with the training duration, and that the training loss decays empirically at a $1/\sqrt{t}$ rate, consistent with the rates of Theorems 3.1 and 3.2. The modified step size itself is not analyzed and no convergence guarantees are provided. It is therefore worth investigating the theoretical properties of such more advanced SPS-type rules and our analysis is a first step in this direction.

B Proofs of Convergence Guarantees

This section contains the proofs of Theorems 3.1 and 3.2. We first collect the auxiliary lemmas used in both proofs, then present each theorem proof in turn. Throughout this section, $(X)_+^2$ denotes $\big((X)_+\big)^2$, and $\nabla f(\boldsymbol{y})$ without a subscript denotes the gradient of the full objective f.

B.1 Auxiliary Lemmas

Lemma B.1 (Extended Titu's Lemma)**.** For any random variable X and positive-valued random variable Y, it holds

$$\mathbb{E}\left[\frac{(X)_+^2}{Y}\right] \geq \frac{(\mathbb{E}\left[X\right])_+^2}{\mathbb{E}\left[Y\right]}. \tag{9}$$

In addition, for any numbers $a_0, \ldots, a_k$ and positive numbers $b_0, \ldots, b_k$, we have

$$\sum_{t=0}^{k} \frac{(a_t)_+^2}{b_t} \geq \frac{\left(\sum_{t=0}^{k} a_t\right)_+^2}{\sum_{t=0}^{k} b_t}. \tag{10}$$

Lemma B.2. If $f_{\boldsymbol{\zeta}}$ is convex for every $\boldsymbol{\zeta}$, and we use the learning rate SF-SPS$_+$ we have that

$$\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2 \leq \|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2 - \frac{\left(f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t) - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) + \beta\left\langle \nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\right\rangle\right)_+^2}{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2}. \tag{11}$$

As a consequence we also have that $\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|$, $\|\boldsymbol{x}_t - \boldsymbol{x}_\star\|$ and $\|\boldsymbol{y}_t - \boldsymbol{x}_\star\|$ are less than $\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$. Furthermore, taking expectation we have that

$$\mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] \leq \mathbb{E}\left[\|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2\right] - \frac{\left(\mathbb{E}\left[f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star) + \beta\left\langle \nabla f(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\right\rangle\right]\right)_+^2}{\mathbb{E}\left[\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2\right]}. \tag{12}$$

Proof. Inserting SF-SPS$_+$ into (4) gives the first result, which also shows that $\|\boldsymbol{z}_t - \boldsymbol{x}_\star\| \leq \|\boldsymbol{z}_0 - \boldsymbol{x}_\star\| = \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$. Since $\boldsymbol{x}_{t+1}$ is a convex combination of $\boldsymbol{x}_t$ and $\boldsymbol{z}_t$, we have that

$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_\star\| \leq (1 - c_{t+1})\|\boldsymbol{x}_t - \boldsymbol{x}_\star\| + c_{t+1}\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|$$

from which we can use induction to show $\|\boldsymbol{x}_t - \boldsymbol{x}_\star\| \leq \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$. Furthermore, since $\boldsymbol{y}_t$ is a convex combination of $\boldsymbol{z}_{t-1}$ and $\boldsymbol{x}_t$, it also follows by induction that $\|\boldsymbol{y}_t - \boldsymbol{x}_\star\| \leq \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$.

Taking conditional expectation over (11) given $\boldsymbol{x}_t$ and $\boldsymbol{z}_{t-1}$ and using Lemma B.1 gives

$$\mathbb{E}_t[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2] \leq \|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2 - \frac{\left(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star) + \beta\langle \nabla f(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle\right)_+^2}{\mathbb{E}_t[\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2]}. \tag{13}$$

Finally, taking total expectation over (13), and using the law of total expectation and Lemma B.1 again, yields (12). □

Next we develop the Bregman viewpoint of this method.

Lemma B.3. Let $\lambda = \frac{\beta}{1-\beta}$. It follows that

$$\begin{aligned} f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star) + \beta\left\langle \nabla f(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\right\rangle = (1+\lambda)(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)) \\ - \lambda(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) \\ + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t), \end{aligned} \tag{14}$$

where $B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)$ is the Bregman divergence of f that is

$$B_f(\boldsymbol{x}, \boldsymbol{y}) := f(\boldsymbol{x}) - f(\boldsymbol{y}) - \langle \nabla f(\boldsymbol{y}), \boldsymbol{x} - \boldsymbol{y} \rangle .$$

Proof. Using $\boldsymbol{z}_{t-1} - \boldsymbol{x}_t = \frac{1}{1-\beta}(\boldsymbol{y}_t - \boldsymbol{x}_t)$ which follows from the $\boldsymbol{y}_t$ update in Schedule-Free SGD gives

$$\begin{aligned} f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star) + \beta \langle \nabla f(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle &= f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star) - \frac{\beta}{1-\beta} \langle \nabla f(\boldsymbol{y}_t), \boldsymbol{x}_t - \boldsymbol{y}_t \rangle \\ &= (1+\lambda)(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) \\ &\quad + \lambda\big(f(\boldsymbol{x}_t) - f(\boldsymbol{y}_t) - \langle \nabla f(\boldsymbol{y}_t), \boldsymbol{x}_t - \boldsymbol{y}_t \rangle\big). \end{aligned}$$

□

Lemma B.4. Let $c_t = 1/(t+1)$. Initializing $\boldsymbol{z}_{-1} = \boldsymbol{x}_0$, it follows that

$$\begin{aligned} \mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] \leq \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|^2 \\ - \frac{\big((t+1)\mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right] + \lambda \sum_{k=0}^t \mathbb{E}\left[B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right]\big)_+^2}{\sum_{k=0}^t \mathbb{E}\left[\|\nabla f_{\boldsymbol{\zeta}_k}(\boldsymbol{y}_k)\|^2\right]}. \end{aligned} \tag{15}$$

Proof. Using (14) in (12) gives

$$\begin{aligned} \mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] \leq \mathbb{E}\left[\|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2\right] \\ - \frac{\big(\mathbb{E}\left[(1+\lambda)(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)\right]\big)_+^2}{\mathbb{E}\left[\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2\right]}. \end{aligned} \tag{16}$$

Therefore, unrolling (16) gives

$$\mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] \leq \|\boldsymbol{z}_{-1} - \boldsymbol{x}_\star\|^2 - \sum_{k=0}^t \frac{(a_k)_+^2}{b_k},$$

where we define $a_k := \mathbb{E}\left[(1+\lambda)(f(\boldsymbol{y}_k) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_k) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right]$ and $b_k = \mathbb{E}\left[\|\nabla f_{\boldsymbol{\zeta}_k}(\boldsymbol{y}_k)\|^2\right]$. From Lemma B.1, we know that

$$\sum_{k=0}^t \frac{(a_k)_+^2}{b_k} \geq \frac{\big(\sum_{k=0}^t a_k\big)_+^2}{\sum_{k=0}^t b_k}.$$

Therefore, we have that

$$\begin{aligned} \mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] \leq \|\boldsymbol{z}_{-1} - \boldsymbol{x}_\star\|^2 \\ - \frac{\big(\sum_{k=0}^t \mathbb{E}\left[(1+\lambda)(f(\boldsymbol{y}_k) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_k) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right]\big)_+^2}{\sum_{k=0}^t \mathbb{E}\left[\|\nabla f_{\boldsymbol{\zeta}_k}(\boldsymbol{y}_k)\|^2\right]}. \end{aligned} \tag{17}$$

To finish the proof of convergence, we need to write $\boldsymbol{y}_t$ as a combination of $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ so that we can telescope. To this end note that

$$\boldsymbol{z}_{t-1} = \frac{1}{c_t}\boldsymbol{x}_t + \left(1 - \frac{1}{c_t}\right)\boldsymbol{x}_{t-1}.$$

Substituting this into the $\boldsymbol{y}_t$ update in Schedule-Free SGD gives

$$\boldsymbol{y}_t = (1-\beta)\left(\frac{1}{c_t}\boldsymbol{x}_t + \left(1 - \frac{1}{c_t}\right)\boldsymbol{x}_{t-1}\right) + \beta \boldsymbol{x}_t$$

$$= \left((1-\beta)\left(\frac{1}{c_t}-1\right)+1\right)\boldsymbol{x}_t - (1-\beta)\left(\frac{1}{c_t}-1\right)\boldsymbol{x}_{t-1}.$$

Let $\rho_t := (1-\beta)\left(\frac{1}{c_t}-1\right)$. Isolating $\boldsymbol{x}_t$ in the above we have that it can be expressed as a convex combination between $\boldsymbol{y}_t$ and $\boldsymbol{x}_{t-1}$ given by

$$\boldsymbol{x}_t = \frac{1}{1+\rho_t}\boldsymbol{y}_t + \frac{\rho_t}{1+\rho_t}\boldsymbol{x}_{t-1}. \tag{18}$$

Using the convexity of f we have that

$$f(\boldsymbol{x}_t) \leq \frac{1}{1+\rho_t}f(\boldsymbol{y}_t) + \frac{\rho_t}{1+\rho_t}f(\boldsymbol{x}_{t-1}). \tag{19}$$

Re-arranging and isolating $f(\boldsymbol{y}_t)$ gives

$$f(\boldsymbol{y}_t) \geq (1+\rho_t)f(\boldsymbol{x}_t) - \rho_t f(\boldsymbol{x}_{t-1}). \tag{20}$$

Using the above we have that

$$\begin{aligned}(1+\lambda)(f(\boldsymbol{y}_t)-f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) &\geq (1+\lambda)(1+\rho_t)(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) \\ &\quad - (1+\lambda)\rho_t(f(\boldsymbol{x}_{t-1})-f(\boldsymbol{x}_\star))) \\ &\quad - \lambda(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) \\ &= (1+(1+\lambda)\rho_t)(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) \\ &\quad - (1+\lambda)\rho_t(f(\boldsymbol{x}_{t-1})-f(\boldsymbol{x}_\star)))\end{aligned}$$

Substituting back $\rho_t := (1-\beta)\left(\frac{1}{c_t}-1\right)$ and $1+\lambda = \frac{1}{1-\beta}$ in the above and using that $c_t = 1/(t+\frac{1}{c_0})$ gives

$$\begin{aligned}&(1+\lambda)(f(\boldsymbol{y}_t)-f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) \\ &\quad \geq \left(\frac{1}{c_t}\right)(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) - \left(\frac{1}{c_t}-1\right)(f(\boldsymbol{x}_{t-1})-f(\boldsymbol{x}_\star)) \\ &\quad = \left(t+\frac{1}{c_0}\right)(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) - \left(t-1+\frac{1}{c_0}\right)(f(\boldsymbol{x}_{t-1})-f(\boldsymbol{x}_\star)).\end{aligned}$$

Using the above we have that

$$\begin{aligned}&\sum_{k=0}^{t}\big((1+\lambda)(f(\boldsymbol{y}_k)-f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_k)-f(\boldsymbol{x}_\star))\big) \\ &\quad \geq f(\boldsymbol{x}_0)-f(\boldsymbol{x}_\star) + \sum_{k=1}^{t}\left(\left(k+\frac{1}{c_0}\right)(f(\boldsymbol{x}_k)-f(\boldsymbol{x}_\star)) - \left(k-1+\frac{1}{c_0}\right)(f(\boldsymbol{x}_{k-1})-f(\boldsymbol{x}_\star))\right) \\ &\quad = f(\boldsymbol{x}_0)-f(\boldsymbol{x}_\star) + \left(t+\frac{1}{c_0}\right)(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)) - \frac{1}{c_0}(f(\boldsymbol{x}_0)-f(\boldsymbol{x}_\star)) \\ &\quad = (t+1)(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)).\end{aligned}$$

Inserting this in (17), together with the monotonicity of the positive part and the initialization that $\boldsymbol{z}_{-1} = \boldsymbol{x}_0$, gives

$$\begin{aligned}\mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] &\leq \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|^2 \\ &\quad - \frac{\left((t+1)\mathbb{E}\left[f(\boldsymbol{x}_t)-f(\boldsymbol{x}_\star)\right] + \lambda\sum_{k=0}^{t}\mathbb{E}\left[B_f(\boldsymbol{x}_k,\boldsymbol{y}_k)\right]\right)_+^2}{\sum_{k=0}^{t}\mathbb{E}\left[\|\nabla f_{\boldsymbol{\zeta}_k}(\boldsymbol{y}_k)\|^2\right]}.\end{aligned} \tag{21}$$

□

B.2 Proof of Theorem 3.1 (Oracle Step Size)

Proof. Since $\|\boldsymbol{y}_k - \boldsymbol{x}_\star\| \leq \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$ we have that $\mathbb{E}\left[\|\nabla f_{\boldsymbol{\zeta}_k}(\boldsymbol{y}_k)\|^2\right] \leq G^2$ and re-arranging (15) gives

$$\begin{aligned}&\left((t+1)\mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right] + \lambda \sum_{k=0}^{t} \mathbb{E}\left[B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right]\right)_+^2 \\ &\quad\leq G^2(t+1)\left(\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|^2 - \mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right]\right) \leq G^2(t+1)\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|^2.\end{aligned}$$

Since the term on the left is always positive we can drop the positive part, taking square roots, and dividing through by $t+1$ gives

$$\mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right] + \frac{\lambda}{t+1}\sum_{k=0}^{t}\mathbb{E}\left[B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right] \leq \frac{G\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|}{\sqrt{t+1}}.$$

Inserting back $\lambda = \beta/(1-\beta)$ gives

$$\mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right] \leq \mathbb{E}\left[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)\right] + \frac{1}{t+1}\frac{\beta}{1-\beta}\sum_{k=0}^{t}\mathbb{E}\left[B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right] \leq \frac{G\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|}{\sqrt{t+1}}.$$

Finally, we can drop the positive terms given by the Bregman divergences $\mathbb{E}\left[B_f(\boldsymbol{x}_k, \boldsymbol{y}_k)\right]$, giving the final desired result. □

B.3 Proof of Theorem 3.2 (Safeguarded Step Size)

Proof. Let $q := f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t) + \beta\langle\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle$. We have

$$\begin{aligned}&\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2 - \|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2 \\ &\leq -2\gamma_t\left[f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t) - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) + \beta\langle\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle\right] + \gamma_t^2\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2 \\ &= -\frac{2[q - \ell^*_{\boldsymbol{\zeta}_t}]_+[q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)]}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} + \frac{[q - \ell^*_{\boldsymbol{\zeta}_t}]_+^2}{(\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\})^2}\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2 \\ &= \frac{-2(q - \ell^*_{\boldsymbol{\zeta}_t})_+ \cdot (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star))}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \\ &\quad + \frac{(q - \ell^*_{\boldsymbol{\zeta}_t})_+^2}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \cdot \frac{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \\ &\leq \frac{-2(q - \ell^*_{\boldsymbol{\zeta}_t})_+ \cdot (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star))}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} + \frac{(q - \ell^*_{\boldsymbol{\zeta}_t})_+^2}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \\ &= \frac{-2(q - \ell^*_{\boldsymbol{\zeta}_t})_+ \cdot (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)) + (q - \ell^*_{\boldsymbol{\zeta}_t})_+^2}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \\ &\overset{(\star)}{\leq} \frac{(f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) - \ell^*_{\boldsymbol{\zeta}_t})^2 - (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star))_+^2}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \\ &= -\frac{(f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t) - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) + \beta\langle\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle)_+^2}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \\ &\quad + \frac{[f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) - \ell^*_{\boldsymbol{\zeta}_t}]^2}{\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\}} \\ &\overset{(\star\star)}{\leq} -\frac{(f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t) - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) + \beta\langle\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle)_+^2}{\max\{G^2, M\}} + \frac{[f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) - \ell^*_{\boldsymbol{\zeta}_t}]^2}{M}\end{aligned}$$

Let's explain inequality $(\star)$: Note that $\ell^*_{\boldsymbol{\zeta}_t} \leq f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)$ so $q - \ell^*_{\boldsymbol{\zeta}_t} \geq q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)$. Hence if $q - \ell^*_{\boldsymbol{\zeta}_t} \leq 0$ inequality $(\star)$ reduces to the obvious $0 \leq [f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) - \ell^*_{\boldsymbol{\zeta}_t}]^2$. Now assume that $q - \ell^*_{\boldsymbol{\zeta}_t} > 0$. Then

$$-2(q - \ell^*_{\boldsymbol{\zeta}_t})_+ \cdot (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)) + (q - \ell^*_{\boldsymbol{\zeta}_t})_+^2 = -2(q - \ell^*_{\boldsymbol{\zeta}_t})(q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)) + (q - \ell^*_{\boldsymbol{\zeta}_t})^2$$

$$
\begin{aligned}
&= (q - \ell^*_{\boldsymbol{\zeta}_t} - (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star)))^2 - (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star))^2 \\
&= (f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) - \ell^*_{\boldsymbol{\zeta}_t})^2 - (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star))^2 \\
&\leq (f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star) - \ell^*_{\boldsymbol{\zeta}_t})^2 - (q - f_{\boldsymbol{\zeta}_t}(\boldsymbol{x}_\star))_+^2.
\end{aligned}
$$

Inequality $(\star\star)$ follows from $\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\} \geq M$ and $\max\{\|\nabla f_{\boldsymbol{\zeta}_t}(\boldsymbol{y}_t)\|^2, M\} \leq \max\{G^2, M\}$ because $f_{\boldsymbol{\zeta}_t}$ is G-Lipschitz.

Taking conditional and then total expectation, and using Lemma B.1, the tower property, Lemma B.3, and the definition of σ^2 in (6), we get

$$
\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] - \mathbb{E}\left[\|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2\right] \\
&\leq -\frac{\mathbb{E}\left[(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star) + \beta\langle\nabla f(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle)_+^2\right]}{\max\{G^2, M\}} + \frac{\sigma^4}{M} \\
&= -\frac{\mathbb{E}\left[((1+\lambda)(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t))_+^2\right]}{\max\{G^2, M\}} + \frac{\sigma^4}{M},
\end{aligned}
$$

where $\lambda = \beta/(1-\beta)$, so

$$
\begin{aligned}
&\mathbb{E}\left[((1+\lambda)(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t))_+^2\right] \\
&\leq \max\{G^2, M\}\mathbb{E}\left[\|\boldsymbol{z}_{t-1} - \boldsymbol{x}_\star\|^2\right] - \max\{G^2, M\}\mathbb{E}\left[\|\boldsymbol{z}_t - \boldsymbol{x}_\star\|^2\right] + \frac{\max\{G^2, M\}}{M}\sigma^4.
\end{aligned}
$$

Now let $\Delta_t = (1+\lambda)(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)$. Summing for $t = 0, \ldots, T-1$, telescoping, and using Jensen's inequality gives

$$
\begin{aligned}
&\frac{\max\{G^2, M\}\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|^2}{T} + \frac{\max\{G^2, M\}}{M}\sigma^4 \\
&\geq \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[(\Delta_t)_+^2\right] \\
&\geq \left(\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\Delta_t\right]\right)_+^2,
\end{aligned}
$$

which means that

$$
\begin{aligned}
\left(\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\Delta_t\right]\right)_+ &\leq \sqrt{\frac{\max\{G^2, M\}\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|^2}{T} + \frac{\max\{G^2, M\}}{M}\sigma^4} \\
&\leq \frac{\sqrt{\max\{G^2, M\}}\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|}{\sqrt{T}} + \sqrt{\frac{\max\{G^2, M\}}{M}}\sigma^2.
\end{aligned}
$$

As in (18), we have $\boldsymbol{x}_t = \frac{1}{1+\rho_t}\boldsymbol{y}_t + \frac{\rho_t}{1+\rho_t}\boldsymbol{x}_{t-1}$, and using convexity and solving for $f(\boldsymbol{y}_t)$ as in (20) we get

$$
f(\boldsymbol{y}_t) \geq (1+\rho_t)f(\boldsymbol{x}_t) - \rho_t f(\boldsymbol{x}_{t-1}),
$$

where $\rho_t = (1-\beta)(\frac{1}{c_t} - 1)$. Now we have

$$
\begin{aligned}
&(1+\lambda)[f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)] - \lambda[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)] + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t) \\
&\geq (t+1)[f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)] - t[f(\boldsymbol{x}_{t-1}) - f(\boldsymbol{x}_\star)] + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t),
\end{aligned}
$$

thus

$$
\begin{aligned}
\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\Delta_t\right] &= \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[(1+\lambda)(f(\boldsymbol{y}_t) - f(\boldsymbol{x}_\star)) - \lambda(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)\right] \\
&\geq \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1}(t+1)(f(\boldsymbol{x}_t) - f(\boldsymbol{x}_\star)) - t(f(\boldsymbol{x}_{t-1}) - f(\boldsymbol{x}_\star)) + \lambda B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)\right]
\end{aligned}
$$

$$= \mathbb{E}\left[f(\boldsymbol{x}_{T-1}) - f(\boldsymbol{x}_\star)\right] + \frac{\lambda}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)\right]$$
$$\geq 0.$$

Hence

$$\left(\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\Delta_t\right]\right)_+ \geq \mathbb{E}\left[f(\boldsymbol{x}_{T-1}) - f(\boldsymbol{x}_\star)\right] + \frac{\lambda}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)\right],$$

thus

$$\mathbb{E}\left[f(\boldsymbol{x}_{T-1}) - f(\boldsymbol{x}_\star)\right] + \frac{\beta}{1-\beta}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[B_f(\boldsymbol{x}_t, \boldsymbol{y}_t)\right]$$
$$\leq \frac{\sqrt{\max\{G^2, M\}}\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|}{\sqrt{T}} + \sqrt{\frac{\max\{G^2, M\}}{M}}\sigma^2,$$

which holds for every $T \geq 1$. Setting $T = t + 1$ gives (7), as wanted. □

C Additional Experimental Results

C.1 Pseudocodes

For completeness, we provide the pseudocode for the Adam-based variants used in our experiments; the SGD-based method is given in Algorithm 1.

Algorithm 2 Schedule-Free Adam with SF-Adam-SPS$_+$ or SF-Adam-SPS$_{\text{safe}}$

Input: $\boldsymbol{z}_{-1} = \boldsymbol{x}_0 \in \mathbb{R}^d$, $\beta \in [0,1)$, $\beta_2 \in [0,1)$, $\epsilon > 0$, lower bounds $\ell^*_{\zeta_t}$, safeguard $M > 0$
$\boldsymbol{v}_{-1} = \boldsymbol{0}$
for $t = 0$ **to** $T-1$ **do**
  $\boldsymbol{y}_t = (1-\beta)\boldsymbol{z}_{t-1} + \beta \boldsymbol{x}_t$
  Sample ζ_t; compute $\nabla f_{\zeta_t}(\boldsymbol{y}_t)$
  $\boldsymbol{v}_t = \beta_2 \boldsymbol{v}_{t-1} + (1-\beta_2)\left(\nabla f_{\zeta_t}(\boldsymbol{y}_t)\right)^2$ ▷ Element-wise square
  $\boldsymbol{D}_t = \operatorname{diag}\left(\sqrt{\boldsymbol{v}_t/(1-\beta_2^{t+1})} + \epsilon\right)$ ▷ Adam preconditioner
  Option I: $\gamma_t = \dfrac{[f_{\zeta_t}(\boldsymbol{y}_t) - f_{\zeta_t}(\boldsymbol{x}_\star) + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle]_+}{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2_{\boldsymbol{D}_t^{-1}}}$ ▷ SF-Adam-SPS$_+$
  or
  Option II: $\gamma_t = \dfrac{\left[f_{\zeta_t}(\boldsymbol{y}_t) - \ell^*_{\zeta_t} + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t\rangle\right]_+}{\max\{\|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2_{\boldsymbol{D}_t^{-1}}, M\}}$ ▷ SF-Adam-SPS$_{\text{safe}}$
  $\boldsymbol{z}_t = \boldsymbol{z}_{t-1} - \gamma_t \boldsymbol{D}_t^{-1} \nabla f_{\zeta_t}(\boldsymbol{y}_t)$
  $\boldsymbol{x}_{t+1} = (1-c_{t+1})\boldsymbol{x}_t + c_{t+1}\boldsymbol{z}_t, \quad c_{t+1} = \frac{1}{t+1}$
end for
Return: $\boldsymbol{x}_T$

C.2 Image Classification with SF-SPS$_{\text{safe}}$

This section complements Section 5.1 with the complete training protocol, the final training losses, and the safeguard diagnostics for the image classification experiments. Recall that these experiments evaluate the SGD-based method analyzed in Section 3, namely Algorithm 1 with SF-SPS$_{\text{safe}}$, run verbatim with the harmonic averaging weights $c_{t+1} = 1/(t+1)$.

Protocol. We train ResNet-20 and ResNet-32 [He et al., 2016] with batch normalization on CIFAR-10 and CIFAR-100 [Krizhevsky, 2009] for 200 epochs with batch size 128, using the standard augmentation of random crops with 4-pixel padding and random horizontal flips, per-channel normalization, and random reshuffling at every epoch; the loss is the cross-entropy. Both methods use the momentum parameter $\beta = 0.9$, no warmup, and no weight decay. The baseline is Schedule-Free SGD with a constant learning rate, swept over the six values $\gamma \in \{0.25, 0.5, 1, 2, 5, 10\}$; we report the best and the worst member of the sweep. SF-SPS$_{\text{safe}}$ uses the same defaults as in Section 5.3, namely $\ell^*_{\zeta_t} = 0$ and either the EMA safeguard of Section 4.2 with $\beta_M = 0.99$ or a fixed $M \in \{1, 2, 5, 10, 20, 50\}$. Following Defazio et al. [2024], we evaluate the averaged iterate $\boldsymbol{x}_t$ after every epoch, recomputing the batch normalization statistics at $\boldsymbol{x}_t$ on 50 training batches before each evaluation. We report the best test accuracy over the 200 evaluations and the training loss averaged over the final epoch. Each run uses a single NVIDIA RTX 6000 Ada Generation GPU.

Results. The complete results are presented in Table 4. The test-accuracy block coincides with Table 3 and is discussed in Section 5.1; in addition, we report the training loss averaged over the final epoch for every configuration.

Sensitivity to the safeguard. As we saw in Section 5.1, across $M \in \{1, 2, 5, 10\}$ and the EMA safeguard the test accuracy varies by at most 1.5 points in each of the four settings. Table 5 reports how often the safeguard is active. For $M \leq 2$ the safeguard essentially never binds and the step size is the unclipped Polyak step. As M grows, the safeguard binds in an increasing

Table 4: Test accuracy (%) and final training loss after 200 epochs on CIFAR-10/100. Schedule-Free SGD uses a constant learning rate swept over six values; we report the best and the worst member of the sweep, with the corresponding learning rate in parentheses. SF-SPS$_{\text{safe}}$ uses $\ell^*_{\zeta_t} = 0$, and either the EMA safeguard with $\beta_M = 0.99$ or a fixed M.

	Schedule-Free SGD (LR sweep)		SF-SPS$_{\text{safe}}$ (no sweep)						
Setting	best (LR)	worst (LR)	EMA M	M=1	M=2	M=5	M=10	M=20	M=50
Test accuracy (%)									
CIFAR-10 / ResNet-20	88.70 (1)	82.02 (10)	86.26	86.50	86.53	86.06	85.08	84.04	81.21
CIFAR-10 / ResNet-32	89.35 (0.5)	81.40 (10)	85.63	86.07	86.06	86.11	86.05	84.77	81.67
CIFAR-100 / ResNet-20	61.73 (1)	55.74 (10)	59.84	59.48	60.44	60.60	59.21	59.18	55.66
CIFAR-100 / ResNet-32	60.91 (0.5)	50.35 (10)	58.75	59.52	59.96	58.69	59.82	57.48	56.47
Final training loss									
CIFAR-10 / ResNet-20	0.017 (1)	—	0.159	0.136	0.136	0.146	0.186	0.283	0.439
CIFAR-10 / ResNet-32	0.009 (0.5)	—	0.142	0.129	0.129	0.127	0.150	0.219	0.400
CIFAR-100 / ResNet-20	0.315 (1)	—	0.588	0.543	0.554	0.585	0.651	0.784	1.164
CIFAR-100 / ResNet-32	0.136 (0.5)	—	0.500	0.403	0.409	0.458	0.486	0.615	0.947

Table 5: Safeguard diagnostics for SF-SPS$_{\text{safe}}$ on CIFAR-10/100: fraction of logged steps (%) in which the safeguard bound the denominator, i.e., $M \geq \|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2$ (top), and in which the ReLU $[\cdot]_+$ set the step size to zero (bottom).

Setting	EMA M	M=1	M=2	M=5	M=10	M=20	M=50
Safeguard active (%)							
CIFAR-10 / ResNet-20	55.0	0.0	0.0	14.7	54.8	98.7	100.0
CIFAR-10 / ResNet-32	53.9	0.0	0.0	1.8	34.5	87.9	100.0
CIFAR-100 / ResNet-20	52.5	0.8	1.1	3.8	20.3	75.0	100.0
CIFAR-100 / ResNet-32	52.0	0.0	0.3	1.0	5.6	25.3	99.6
ReLU active (%)							
CIFAR-10 / ResNet-20	0.9	1.5	1.4	1.2	0.5	0.1	0.0
CIFAR-10 / ResNet-32	1.3	1.9	1.9	2.2	1.7	0.5	0.0
CIFAR-100 / ResNet-20	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CIFAR-100 / ResNet-32	0.0	0.0	0.0	0.0	0.0	0.0	0.0

fraction of the steps, reaching essentially every step for $M = 50$, so that the step size degenerates to $[f_{\zeta_t}(\boldsymbol{y}_t) + \beta \langle \nabla f_{\zeta_t}(\boldsymbol{y}_t), \boldsymbol{z}_{t-1} - \boldsymbol{x}_t \rangle]_+ / M$; accordingly, the accuracy drops by 1 to 5 points for $M \in \{20, 50\}$. The EMA safeguard binds in roughly half of the steps in every setting and removes the need to choose M altogether. Finally, the ReLU in the numerator is active in at most 2.3% of the steps on CIFAR-10 and never on CIFAR-100, consistent with the remark in Section 2.3 that a negative numerator rarely occurs in practice.

C.3 Black-Box Distillation details

Experiment	`tiny_shakespeare`
Teacher model	`gpt2-medium`
Student hidden size	768
Student transformer layers	4
Student attention heads	8
Student vocabulary size	50257
Batch size	4
Context length	512 tokens
Tokens per training step	4096
Learning rate schedule	constant
(β_1, β_2)	(0.9, 0.999)

Table 6: Model configuration and training setup for distillation on `tiny_shakespeare`.

Mixed precision training was enabled using `bfloat16` for efficiency. The student model utilized flash attention [Dao et al., 2022]. The full details of the architecture and hyperparameters we used are in Table 6.

C.4 Pretraining GPT-2 Decoder Experiment Details

We follow the GPT-2 architecture [Radford et al., 2019] and train on FineWeb 10B [Penedo et al., 2024]. As in Defazio et al. [2024], gradient clipping is disabled for Schedule-Free and SF-Adam-SPS$_{\text{safe}}$ runs.

Architecture. Table 7 summarizes the model configurations. All models use flash attention, a vocabulary of 50,257 tokens, and a context length of 1,024.

Table 7: Model architectures and hardware.

Hyperparameter	GPT-small	GPT-medium
Parameters	124M	355M
Num. embeddings	768	1,024
Num. layers	12	24
Num. heads	12	16
GPUs	4×H100	4×H100

Training. Table 8 lists the common training hyperparameters shared across all models.

Table 8: Common training hyperparameters.

Hyperparameter	Value
Dataset	FineWeb 10B
Total tokens	10.25B
Context length	1,024
Global batch size	480 sequences (491,520 tokens)
Per-GPU batch size	12
Gradient accum. steps	10
Optimizer steps	20,856
Weight decay	0.1
Dropout	0.0
Gradient clipping	1.0 (AdamW only)

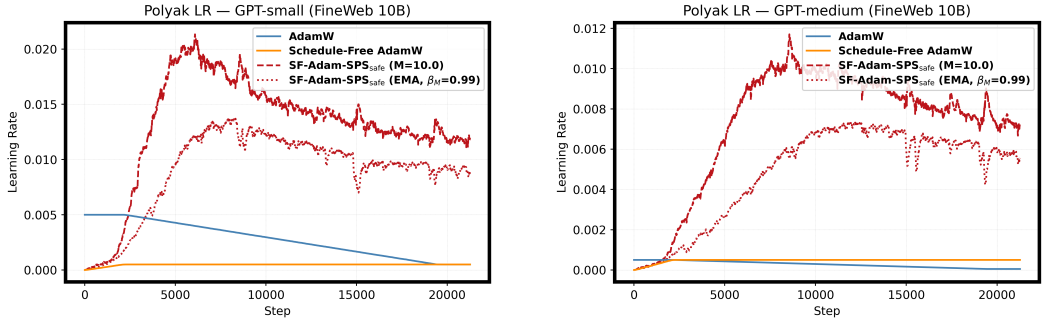
Optimizers. Table 9 lists the optimizer-specific hyperparameters. For AdamW and Schedule-Free Adam we sweep the learning rate over six values and report the best validation loss. The Polyak step size is determined adaptively and requires no LR sweep, we therefore plot the final training and validation losses as a constant horizontal line to compare against the performance of the baselines for all base learning rate settings.

Table 9: Optimizer hyperparameters.

Hyperparameter	AdamW	SF-Adam	SF-Adam-SPS$_{\text{safe}}$ (const M)	SF-Adam-SPS$_{\text{safe}}$ (EMA M)
β_1	0.9	0.9	0.9	0.9
β_2	0.95	0.98	0.98	0.98
Warmup steps	—	2,000	2,000	2,000
LR schedule	linear decay	constant	constant	constant
LR sweep	{5e-6, 5e-5, 5e-4, 5e-3, 5e-2, 5e-1}	{5e-6, 5e-5, 5e-4, 5e-3, 5e-2, 5e-1}	—	—
M	—	—	10	—
β_M (EMA coeff.)	—	—	—	0.99
$\ell^*_{\zeta_t}$	—	—	0.0	0.0

Visualization of Polyak Step Sizes Figure 3a shows the adaptive step-sizes computed by SF-Adam-SPS$_{\text{safe}}$ over the course of training, compared against the fixed learning rates of AdamW and

Schedule-Free AdamW tuned for best final validation loss. The Polyak step-sizes are large early in training when the loss gap is large, and decay naturally as the model converges, recovering a schedule-like decay without any explicit schedule.



(a) GPT2-small (b) GPT2-medium

Figure 3: Adaptive step sizes computed by SF-Adam-SPS$_{safe}$ over the course of training, along with the fixed learning rates of AdamW and Schedule-Free AdamW (picked for best final validation loss).

Activation of Safeguard M Figure 4 shows M plotted against the preconditioned gradient norms throughout training. When M dominates the denominator the safeguard is active, preventing excessively large step-sizes. With EMA-based tracking, M adapts to the scale of the gradient norms, activating more selectively than a fixed M. As demonstrated by Figure 2, in the underparameterized regime the safeguard is not necessary to improve performance and the two methods for selecting M are comparable in terms of training dynamics and generalization performance.

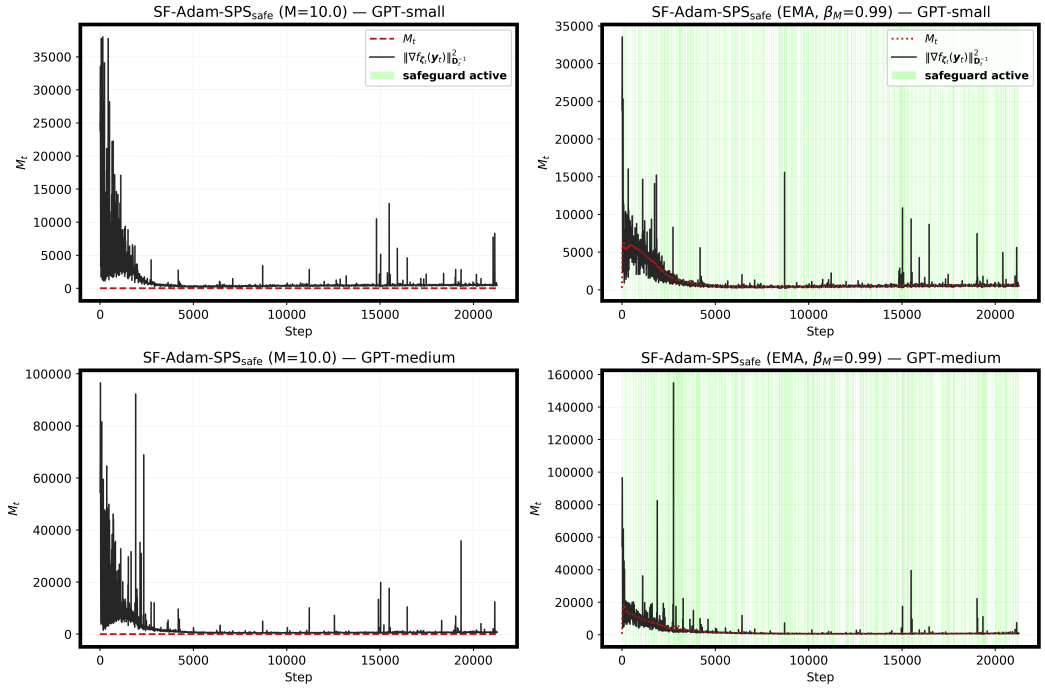


Figure 4: Safeguard parameter M for SF-Adam-SPS$_{safe}$ plotted against the preconditioned gradient norms. A vertical green line indicates that $M \geq \|\nabla f_{\zeta_t}(\boldsymbol{y}_t)\|^2_{\boldsymbol{D}_t^{-1}}$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction (Section 1) state three contributions: (i) derivation of Polyak-type step sizes for Schedule-Free SGD (Sections 2.2 and 2.3), (ii) $\mathcal{O}(1/\sqrt{t})$ convergence guarantees (Theorems 3.1 and 3.2), and (iii) empirical evaluation on image classification, black-box distillation, and language model pretraining (Section 5). All three are supported in the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses several limitations: the oracle step size SF-SPS$_+$ requires access to $f_{\zeta_t}(\boldsymbol{x}_\star)$ (Section 2.2); the safeguarded variant converges only to a neighborhood controlled by σ^2 (Theorem 3.2); and the convergence analysis does not cover smooth, strongly convex, or nonconvex settings, nor the Adam-based extension.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Theorems 3.1 and 3.2 state all assumptions explicitly (convexity, differentiability, Lipschitz continuity, initialization). Complete proofs are provided in Appendix B, building on auxiliary lemmas (Lemmas B.1–B.4) that are stated and proved in Appendix B.1. All theorems, lemmas, and equations are numbered and cross-referenced.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The algorithms are fully specified in pseudocode (Algorithms 1 and 2). All datasets are publicly available. The image classification protocol is given in Section 5.1 and Appendix C.2, the distillation configuration in Table 6 of Appendix C.3, and the pretraining setup in Appendix C.4, with model architectures in Table 7, common training hyperparameters in Table 8, and optimizer-specific settings and learning rate sweeps in Table 9. The distillation and pretraining setups follow the publicly available configurations of Defazio et al. [2024] and Gower et al. [2025], and code is released at `https://github.com/buchholzmd/GPT-opt`.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The code for the LLM experiments is available at https://github.com/buchholzmd/GPT-opt.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5 describes the setup for image classification, distillation, and pretraining. Full details are in Appendix C: the image classification protocol, including augmentation, batch size, evaluation, and the learning rate sweep, in Appendix C.2; the distillation configuration in Table 6; and the pretraining architectures, common training hyperparameters, and optimizer-specific settings including learning rate sweeps in Tables 7–9. The distillation setup follows Gower et al. [2025] and the pretraining setup follows Defazio et al. [2024].

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: All experiments are single runs without error bars or confidence intervals. This is consistent with standard practice in language model pretraining research, where the computational cost of each run makes multiple repetitions prohibitive and single-seed evaluations are the norm (e.g., Defazio et al. [2024], Radford et al. [2019]).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Table 4 specifies the GPU type ($4\times$H100) used for each model scale. Total tokens processed (10.25B) and number of optimizer steps (20,856) are reported in Table 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

Justification: This work proposes optimization algorithms and evaluates them on publicly available datasets and models. It does not involve human subjects, private data, or dual-use concerns.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: This paper presents foundational research on optimization algorithms for stochastic optimization. It is not tied to a particular application and does not have a direct path to negative societal impacts beyond those inherent to general-purpose optimization methods.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release any new models or datasets. The contribution is an optimization algorithm.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All pretrained models and datasets are cited with their original publications. The models are available under permissive open-source licenses (MIT for GPT-2, Apache 2.0 for GPT-J). Specific license names are not listed in the paper but all assets are used in accordance with their terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new datasets, models, or code packages.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing or research with human subjects.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not used as a component of the core methodology. The paper trains language models as experimental benchmarks, but no LLM is used in the design or analysis of the proposed optimization algorithms.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.