EqWD: Bounded Weight Decay from Standardized Ratio Deviations

Anonymous authors
Paper under double-blind review

Abstract

Weight decay controls parameter scale in modern optimizers, and decoupled implementations expose it as a separate update. Most recipes keep its coefficient fixed while loss-driven update size changes during training. The gradient-to-weight norm ratio tracks that size, but absolute values vary across tensors and mix persistent scale with temporal change. We measure each tensor's deviation from its moving ratio mean and normalize it by a moving residual scale. We use this standardized deviation to drive adaptive decoupled decay, which we call Equilibrium-Driven Weight Decay (EqWD). EqWD increases the base coefficient in proportion to the deviation, changes only radial decay, and leaves the base optimizer's loss update unchanged. Including the current residual in the scale bounds the per-step coefficient; the exponential moving-average (EMA) dynamics also bound accumulated squared modulation over a finite horizon and characterize when the signal vanishes. Across three paired 90-epoch ImageNet-1K ResNet-50 runs, EqWD improves best-epoch Top-1 by 1.049 ± 0.238 points over the reference fixed-decay recipe. The exposure-matched fixed-decay control reduces this margin to 0.255 points, showing that effective decay strength explains much of the reference gain. On 300-epoch ViT-Tiny/ImageNet-1K, EqWD reaches $73.739 \pm 0.060\%$ final Top-1 versus $73.718 \pm 0.113\%$ for fixed decay. On COCO2017 DINO (DETR with Improved DeNoising Anchor Boxes) detection, EqWD raises average precision (AP) by 0.6 points with ResNet-50 and 0.1 points with Swin-T.

1 Introduction

Weight decay controls how strongly a training recipe shrinks parameters relative to its loss-driven updates. Decoupled weight decay makes this choice explicit by applying a radial update separately from the optimizer's gradient transformation (Loshchilov & Hutter, 2019). Although the learning-rate schedule changes the size of this update, a fixed decay coefficient does not respond directly to changes in gradient magnitude or parameter scale. Analyses of normalized networks connect the gradient-to-weight norm ratio to the interaction between learning rate, parameter scale, and decay (Defazio, 2025; Kosson et al., 2024). This connection suggests monitoring the ratio during training, but a controller still needs a reference against which to interpret its changes.

Using the absolute ratio as that reference creates an ambiguity. A large value can reflect a large gradient, a small parameter norm, or a persistent scale difference between tensors. Even a change in the ratio does not identify which of its two components caused it. Moreover, an equal absolute departure can represent a small fluctuation for one tensor and a large shift for another. We therefore compare each tensor's ratio with its own recent history. A moving mean tracks the local level, while a moving residual scale measures the size of departures from that level. Dividing the absolute residual by this scale gives a *standardized ratio deviation* that distinguishes departures from the persistent scale of each tensor.

We use this deviation to construct Equilibrium-Driven Weight Decay (EqWD). At each optimizer step, EqWD updates the tensor's moving mean and residual scale, then increases a base decay coefficient in proportion to the normalized deviation. The controller uses the existing loss gradient and maintains two scalar states per eligible tensor. It preserves the current base-optimizer direction, including momentum or adaptive preconditioning, while changing the radial contribution to the total update. Both upward and downward departures increase decay, so the base coefficient sets a floor on

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

regularization. Equilibrium analyses motivate the monitored ratio; the moving empirical reference determines how EqWD responds during training.

The order of these updates makes the response bounded. The current residual contributes to the exponential moving-average (EMA) scale before EqWD normalizes that residual, yielding an explicit upper bound on the coefficient for every finite ratio trajectory. The bound depends on the scale memory and modulation gain, which therefore control both temporal response and maximum amplification. An analysis of the evolving states also characterizes when modulation vanishes and bounds its accumulation over time. Persistent ratio fluctuations can sustain additional decay even when individual coefficients remain bounded. This distinction between transient peaks and sustained strength motivates our fixed-decay controls.

Our evaluation examines both the resulting accuracy and the source of its changes. On 90-epoch ResNet-50/ImageNet-1K training, EqWD improves best-epoch Top-1 over the reference fixed-decay recipe by 1.049 ± 0.238 points. The mean margin falls to 0.255 points against exposure-matched fixed decay, and searched fixed decay gives the highest CIFAR-100/ResNet-20 accuracy in the tested comparison (§6.2). These controls identify effective decay strength as a major contributor to the CNN gains. Component ablations examine the role of residual normalization (§6.3), while AdamW experiments test the same controller on ViT classification and DINO detection. The 300-epoch ViT-Tiny/ImageNet-1K results remain close to fixed decay; DINO improves overall COCO AP by 0.6 and 0.1 points with ResNet-50 and Swin-T, respectively (§6.4).

Our contributions are:

1. **A bounded adaptive rule.** EqWD turns standardized ratio deviation into tensor-wise decoupled decay, with an explicit update order and a constant amount of state per tensor (§4).
2. **Per-step and cumulative control.** We derive the signal's response to ratio changes, characterize recovery, and bound both instantaneous and accumulated modulation (§5).
3. **An evaluation with strength controls.** We report 90-epoch ImageNet training, searched and exposure-matched fixed decay, component ablations, AdamW classification and detection experiments, and measured overhead (§6).

2 Background

Weight decay regularizes training by shrinking parameters (Loshchilov & Hutter, 2019). For adaptive optimizers, coupling an L_2 penalty to the gradient update can make the effective regularization depend on the preconditioner. Decoupled weight decay separates the radial shrinkage from the loss-driven update and makes its coefficient explicit (Loshchilov & Hutter, 2019). EqWD uses this decoupled branch as its interface. The base optimizer supplies the loss-driven direction, and EqWD changes the radial coefficient.

Momentum SGD and Adam use different update directions and state variables. Adam computes a bias-corrected direction from gradient moment estimates (Kingma & Ba, 2015). Relative norms already play an optimization role in Layer-wise Adaptive Rate Scaling (LARS) (You et al., 2017) and adaptive gradient clipping (AGC) (Brock et al., 2021). These methods use norm information to rescale or clip the loss update. EqWD uses the same observable family in the regularization branch: it measures the historical deviation of a gradient-to-weight ratio and modulates decoupled decay.

The connection between relative norms and decay follows from a simple equilibrium calculation. Consider a momentum-free decoupled step $w_{t+1} = (1 - \gamma\lambda)w_t - \gamma g_t$ with $\langle w_t, g_t \rangle = 0$. Balancing the squared norm gives

$$\frac{\|g_t\|^2}{\|w_t\|^2} = \frac{2\lambda}{\gamma} - \lambda^2 \approx \frac{2\lambda}{\gamma}. \tag{1}$$

The approximation requires $\gamma\lambda$ to be small. Analyses of normalized networks and rotational equilibria study related ratio dynamics under their stated assumptions (Defazio, 2025; Kosson et al., 2024). Equation (1) describes a fixed-coefficient balance and motivates EqWD's observable. EqWD uses it without prescribing a target for every tensor or a decay schedule.

3 Related Work

Adaptive weight decay. Scheduled Weight Decay (SWD) uses gradient-norm information to schedule the decay coefficient (Xie et al., 2023). Cautious Weight Decay (CWD) conditions decay on update–parameter sign alignment (Chen et al., 2026). Constrained Parameter Regularization (CPR) uses an augmented-Lagrangian formulation to enforce parameter constraints (Franke et al., 2024). Weight Norm Control directly regulates parameter magnitudes (Loshchilov, 2023). Dropout changes feature co-adaptation (Hinton et al., 2012). EqWD instead uses a tensor-wise, history-relative ratio deviation. The experiments compare this signal with FixedWD controls.

Decay schedules and adaptive optimization. Learning-rate and decay schedules interact through the product of the step size and decay coefficient; cosine restarts illustrate explicit schedule design (Loshchilov & Hutter, 2017). Adam and related methods can change the relationship between a nominal coefficient and the realized parameter update (Kingma & Ba, 2015; Loshchilov & Hutter, 2019). Optimizer choice, batch size, and regularization also interact in adaptive-gradient training (Wilson et al., 2017; Goyal et al., 2017). EqWD keeps the current base-optimizer direction and changes the separate decay branch.

Norm-based regularization and sharpness. Gradient clipping and layer-wise scaling use parameter and gradient norms to control update magnitudes (You et al., 2017; Brock et al., 2021). Sharpness-Aware Minimization perturbs parameters to seek updates with lower local loss sharpness (Foret et al., 2021); large-batch studies connect optimization dynamics with the geometry of the learned solution (Keskar et al., 2017). Those methods target the loss update or local geometry. EqWD monitors a relative norm signal and applies it to decoupled shrinkage.

Vision architectures and detection. Our evaluation includes residual networks (He et al., 2016), Vision Transformers (Dosovitskiy et al., 2021; Touvron et al., 2021), and DINO detection with a Swin Transformer backbone (Zhang et al., 2023; Liu et al., 2021). These architectures expose different optimizer interfaces and parameter scales to the same decay controller. The comparison tests whether the update can operate across them while leaving the architecture-specific signal unspecified.

4 Equilibrium-Driven Weight Decay

EqWD maintains a local reference and residual scale for each eligible parameter tensor. Let w_t^l denote tensor l, g_t^l the loss gradient supplied to its optimizer step after accumulation and unscaling, and u_t^l the base-optimizer direction, including momentum or preconditioning. When a recipe clips gradients, the controller observes the clipped gradient; Appendix B.1 gives the detection wrapper's order. For parameter groups with different learning rates, γ_t denotes the rate for the group containing tensor l. The base coefficient is λ_0 and the modulation gain is β. We compute the controller quantities before modifying w_t^l. For successful finite-gradient optimizer steps,

$$r_t^l = \frac{\|g_t^l\|_2}{\|w_t^l\|_2 + \varepsilon}, \tag{2}$$

$$\mu_t^l = \alpha_\mu \mu_{t-1}^l + (1-\alpha_\mu) r_t^l, \qquad e_t^l = r_t^l - \mu_t^l, \tag{3}$$

$$v_t^l = \alpha_v v_{t-1}^l + (1-\alpha_v)(e_t^l)^2, \tag{4}$$

$$D_t^l = \max(\sqrt{v_t^l}, \sigma_{\min}) + \varepsilon, \qquad q_t^l = \frac{|e_t^l|}{D_t^l}, \tag{5}$$

$$\lambda_t^l = \lambda_0(1+\beta q_t^l), \qquad w_{t+1}^l = (1-\gamma_t \lambda_t^l) w_t^l - \gamma_t u_t^l. \tag{6}$$

We use $\lambda_0, \beta \geq 0$, $0 \leq \alpha_\mu, \alpha_v < 1$, $\sigma_{\min} > 0$, and $\varepsilon = 10^{-8}$. On a tensor's first observed step, we initialize $\mu_t^l = r_t^l$, $v_t^l = 0$, and $q_t^l = 0$.

The state v_t^l tracks the squared residual around the *updated* mean. It is a residual second moment; q_t^l therefore has no calibrated tail probability or anomaly threshold. The floor $\sigma_{\min}$ limits sensitivity when the residual scale becomes small, and ε protects the ratio denominator.

EqWD uses the magnitude of a ratio departure because its sign alone does not identify excessive regularization. For positive norms, changes in the unstabilized ratio satisfy $\Delta \log r = \Delta \log \|g\| -$

$\Delta \log \|w\|$. Either the numerator or denominator can cause a ratio increase. EqWD removes this sign and applies a nonnegative surcharge to λ_0, preserving the nominal decay floor. A large λ_0 therefore remains large; Section 6.2 tests the resulting stronger-decay confound.

For momentum SGD we use $u_t^l = 0.9u_{t-1}^l + g_t^l$, with zero dampening and no Nesterov update. AdamW supplies its usual bias-corrected adaptive direction with native decay set to zero. The wrapper preserves this loss-driven direction at the current step, while its radial term can change the direction of the total displacement.

Each convolutional or linear weight tensor has its own state. The implementation excludes biases, normalization parameters, and designated embedding/token parameters. We set `layer_aware=false`, which disables the optional layer-type adjustment while tensor-specific EMA states remain active. Mixed-precision runs unscale gradients before computing EqWD statistics, apply manual decay before `GradScaler.step`, and leave a tensor's states unchanged when its gradient is nonfinite. Appendix B gives the full conventions. `FixedWD` denotes the fixed-coefficient comparison to EqWD. The equations describe real-arithmetic updates. Finite precision can round a small shrinkage to zero even when the nominal coefficient satisfies the bound.

Algorithm 1 EqWD on a finite-gradient optimizer step

Require: Base optimizer with built-in decay zero; $\lambda_0, \beta, \alpha_\mu, \alpha_v, \sigma_{\min}$

Accumulate loss gradients; unscale if using mixed precision
Apply gradient clipping if specified by the training recipe
for each eligible tensor w^l with a gradient **do**
  $r^l \leftarrow \|g^l\|_2/(\|w^l\|_2 + \varepsilon)$
  if tensor state is new **then**
    $\mu^l \leftarrow r^l$, $v^l \leftarrow 0$
  else
    $\mu^l \leftarrow \alpha_\mu \mu^l + (1-\alpha_\mu) r^l$
    $v^l \leftarrow \alpha_v v^l + (1-\alpha_v)(r^l - \mu^l)^2$
  end if
  $q^l \leftarrow |r^l - \mu^l|/(\max(\sqrt{v^l}, \sigma_{\min}) + \varepsilon)$
  $w^l \leftarrow [1 - \gamma_t \lambda_0 (1 + \beta q^l)] w^l$
end for
Apply the base optimizer using the already computed gradients

5 Theoretical Analysis

The two EMA recursions control different parts of the decay trajectory. The mean filters ratio changes, and the residual scale limits their normalized magnitude. We analyze both states as they evolve, suppress the tensor index, and write $a = \alpha_\mu$, $b = \alpha_v$, $s = \sigma_{\min}$, and $d = s + \varepsilon$. Index the first observation by zero, so $\mu_0 = r_0$ and $e_0 = v_0 = q_0 = 0$. The subscript on the constant λ_0 denotes base decay.

Proposition 1 (Intrinsic coefficient bound)**.** *For the initialization and parameter ranges in §4, every finite-gradient step satisfies*

$$0 \le q_t \le \frac{1}{\sqrt{1-\alpha_v}}, \qquad \lambda_0 \le \lambda_t \le \lambda_{\max} := \lambda_0 \left(1 + \frac{\beta}{\sqrt{1-\alpha_v}}\right). \tag{7}$$

If $\gamma_t \ge 0$ and $\gamma_t \lambda_{\max} \le 1$, the realized decay branch cannot increase the parameter norm or reverse its sign.

Proof sketch. Since $v_{t-1} \ge 0$, $v_t \ge (1-\alpha_v) e_t^2$. For $e_t \ne 0$, dividing by $\sqrt{v_t}$ gives the bound on q_t; the floor only increases the denominator. For $e_t = 0$, $q_t = 0$. The learning-rate condition gives $0 \le 1 - \gamma_t \lambda_t \le 1$. Appendix A supplies full proofs and edge cases.

The bound links memory and gain: α_v controls both the scale's timescale and the maximum amplification. Using the previous scale v_{t-1} to normalize the current residual would remove this intrinsic bound. We refer to $q_t \to 0$ as modulation recovery.

Proposition 2 (Change response and recovery)**.** *Let $\Delta r_t = r_t - r_{t-1}$. Under the assumptions of Proposition 1,*

$$e_t = a(e_{t-1} + \Delta r_t) = \sum_{k=1}^{t} a^{t-k+1} \Delta r_k. \tag{8}$$

For $0 < a < 1$, recovery satisfies the equivalence

$$q_t \to 0 \quad \Longleftrightarrow \quad e_t \to 0 \quad \Longleftrightarrow \quad \Delta r_t \to 0. \tag{9}$$

If $\lambda_0 \beta > 0$, these conditions also characterize $\lambda_t \to \lambda_0$. For $a = 0$, the modulation is identically zero.

Derivation. The updated mean gives $e_t = a(r_t - \mu_{t-1})$; substituting $r_t - \mu_{t-1} = \Delta r_t + e_{t-1}$ gives Equation (8). The geometric series shows that vanishing increments imply vanishing residuals. The denominator lower bound d then gives recovery. The converse also holds without assuming bounded ratios; Appendix A.2 proves it by first bounding v_t.

Equation (8) shows that the residual filters recent ratio changes and removes a constant ratio level. For $|\Delta r_t| \le L$, it gives the finite-time tracking bound

$$q_t \le \min\left\{ \frac{1}{\sqrt{1-b}}, \frac{a(1-a^t)L}{(1-a)d} \right\}. \tag{10}$$

The signal can vanish while the ratio keeps changing: $r_t = \log(t+1)$ has vanishing modulation because its increments vanish. Here a sets the memory of ratio changes and d limits sensitivity to small residuals.

Theorem 1 (Accumulated modulation)**.** *Under the assumptions of Proposition 1, define*

$$H_s(v) = \begin{cases} v/s^2, & 0 \le v \le s^2, \\ 1 + \log(v/s^2), & v > s^2. \end{cases} \tag{11}$$

For every integer $T \ge 1$,

$$\sum_{t=1}^{T} q_t^2 \le \sum_{t=1}^{T} \min\{v_t/s^2, 1\} + \frac{b}{1-b} H_s(v_T) \le T + \frac{b}{1-b} H_s(v_T), \tag{12}$$

$$\frac{1}{T} \sum_{t=1}^{T} \lambda_t \le \lambda_0 \left[1 + \beta \sqrt{1 + \frac{b H_s(v_T)}{(1-b)T}} \right]. \tag{13}$$

Proof sketch. The scale recursion gives $e_t^2 = v_t + \frac{b}{1-b}(v_t - v_{t-1})$. The concave potential has derivative $H_s'(v) = 1/\max(v, s^2)$, so $(v_t - v_{t-1}) H_s'(v_t) \le H_s(v_t) - H_s(v_{t-1})$. Combining these identities with $q_t^2 \le e_t^2 / \max(v_t, s^2)$ and summing telescopes the potential. Cauchy–Schwarz gives Equation (13).

Theorem 1 separates transient peaks from sustained modulation. If $H_s(v_T) = o(T)$, as for bounded or polynomially growing v_T, the time-average coefficient has upper limit at most $\lambda_0(1+\beta)$. Individual steps retain the larger bound in Proposition 1. An alternating ratio approaches this average bound when its residual amplitude exceeds the stabilizers (Appendix A.6). Appendix A.4 gives learning-rate-weighted bounds on the extra exposure $\sum_t \gamma_t(\lambda_t - \lambda_0)$, connecting the analysis to the strength controls in §6.2.

The bounds describe coefficient dynamics along the ratio sequence produced by training. Appendix A.7 analyzes the coupled parameter, mean, and scale states on a scalar quadratic with plain gradient descent. It gives a sufficient condition linking modulation gain, mean memory, and the denominator floor for local convergence. Momentum, preconditioning, and broader training settings change the base-update terms and require separate analysis.

6 Experiments

We compare reference recipes with fixed-decay strength controls and isolate the modulation components. Classification comparisons use seeds 42, 123, and 456; all means and sample standard deviations below use those runs. The DINO comparison reports AP and AR for both backbones.

6.1 Experimental Setup

We train ResNet-20 on CIFAR-100 for 200 epochs and ResNet-50 on ImageNet-1K for 90 epochs (He et al., 2016). Both use SGD with momentum 0.9 and decoupled decay. CIFAR uses batch size 128, initial learning rate 0.1, and cosine decay. ImageNet uses batch size 256, initial learning rate 0.1, and tenfold step reductions at epochs 30, 60, and 80. Appendix B specifies augmentation, exclusions, schedules, and search budgets.

The additional ViT-Tiny/ImageNet-1K experiment uses AdamW for 300 epochs, 224-pixel inputs, effective batch size 4096, initial learning rate 0.004, five warm-up epochs, and base decay 0.05. We also evaluate DINO (Zhang et al., 2023) on COCO2017 (Lin et al., 2014), using ResNet-50 and Swin-T (Liu et al., 2021) backbones. Appendix B.1 gives the detailed protocols and the reporting scope for these additions.

We report the maximum evaluation accuracy over epochs and the final-epoch accuracy. CIFAR uses its official test split for these evaluations, and ImageNet uses its official validation split. Hyperparameter screening uses these same splits; Appendix B.3 gives the search protocol and comparison scope. Table 3 uses final-epoch Top-1/Top-5 for ViT, with best-epoch Top-1 reported separately in the appendix.

Detection reports overall AP, size-specific AP_s and AP_m, and AR_{100} in Table 3; Appendix C.4 gives AP_{50}, AP_{75}, and the complete average-recall breakdown. AP and AR summarize multiple detection conditions; neither is a single precision or recall value. We express all added scores on the 0–100 scale and report differences in absolute points.

6.2 Fixed-Decay Controls

Table 1 compares EqWD with reference, searched, and exposure-matched fixed decay. This comparison asks whether fixed decay can reproduce the gains through coefficient selection or matched exposure. For CIFAR, an equal-count screen evaluated 12 fixed-decay settings and 12 EqWD settings for 100 epochs with seed 42. We then trained the selected configurations for 200 epochs with three seeds. The selected fixed coefficient, 0.01, gives the highest CIFAR accuracy on both metrics in this comparison.

Table 1: EqWD and fixed-decay controls. Each block fixes the benchmark, backbone, and training length; we report accuracy as mean $\pm$ sample standard deviation (%). Exposure matching uses an estimated parameter- and learning-rate-weighted coefficient.

Method	Decay setting	Best epoch	Final epoch
CIFAR-100 / ResNet-20, 200 epochs			
Reference FixedWD	$\lambda = 5 \cdot 10^{-4}$	65.087 ± 0.093	64.863 ± 0.085
Searched FixedWD	$\lambda = 0.01$	69.180 ± 0.303	68.900 ± 0.397
Exposure-matched FixedWD	estimated λ	67.773 ± 0.035	67.627 ± 0.098
EqWD	$\lambda_0 = 5 \cdot 10^{-4}$	67.627 ± 0.470	67.390 ± 0.537
ImageNet-1K / ResNet-50, 90 epochs			
Reference FixedWD	$\lambda = 10^{-4}$	72.779 ± 0.224	72.727 ± 0.243
Exposure-matched FixedWD	estimated λ	73.573 ± 0.835	73.493 ± 0.770
EqWD	$\lambda_0 = 10^{-4}$	73.828 ± 0.115	73.783 ± 0.075

The exposure control replaces EqWD with a fixed coefficient estimated from its logged tensor-wise coefficients and weighted by parameter counts and the learning-rate schedule. This yields 0.002486235401 on CIFAR and 0.0001790985475 on ImageNet. These estimates match a sampled, globally averaged linear shrinkage exposure (Appendix B.4). On CIFAR, the matched control reaches a slightly higher mean accuracy than EqWD.

On ImageNet, EqWD improves best-epoch accuracy over the reference fixed coefficient by 1.049 ± 0.238 points in paired-seed differences. Against exposure matching, the mean difference falls to 0.255 points; the three differences are $+1.082$, -0.264, and -0.052. The final-epoch metrics retain the reference-recipe improvement but also leave only a small mean margin over exposure matching.

The ImageNet comparison uses reference and exposure-matched coefficients; Appendix B.3 specifies its parameter settings.

The fixed-decay controls attribute part of these gains to effective strength; they leave the contribution of adaptive timing unresolved. Appendix C reports four additional CIFAR recipe variants; their ranking also depends on the selected coefficients and reporting metric.

6.3 Component Ablations

We compare modulation signals to separate the roles of residual scaling, the sign of the departure, and tensor-specific statistics. The study holds $\lambda_0 = 5 \cdot 10^{-4}$, $\beta = 5$, $\alpha_\mu = 0.99$, and $\alpha_v = 0.8$ fixed. Table 2 defines each variant. The mean-normalized variant divides by the local mean, while the global variant computes one ratio from concatenated eligible tensors and uses shared statistics. All other adaptive variants keep tensor-wise statistics.

Table 2: Component variants on CIFAR-100/ResNet-20, 200 epochs. All adaptive variants use the same nominal hyperparameters. Different normalizations change effective decay strength. The full-EqWD and reference-fixed rows reuse the three-seed runs from Table 1.

Variant	Signal replacing q_t	Best-epoch accuracy (%)
Reference fixed	0	65.087 ± 0.093
Absolute deviation	$\lvert e_t \rvert$	65.250 ± 0.250
Mean-normalized	$\lvert e_t \rvert / (\lvert \mu_t \rvert + \varepsilon)$	65.740 ± 0.416
High-side only	$\max(e_t, 0)/D_t$	67.083 ± 0.366
Low-side only	$\max(-e_t, 0)/D_t$	67.383 ± 0.415
Global statistics	$\lvert e_t^{\text{global}} \rvert / D_t^{\text{global}}$	67.600 ± 0.416
Full EqWD	$\lvert e_t \rvert / D_t$	67.627 ± 0.470

Full EqWD exceeds the absolute-deviation and mean-normalized variants by 2.377 ± 0.285 and 1.887 ± 0.055 points in paired differences. These comparisons favor residual-scale normalization at the tested nominal settings. The global variant is within 0.027 points of tensor-wise EqWD; the two one-sided variants also approach the full configuration.

Residual-scale normalization gives the largest observed difference among these fixed-hyperparameter ablations. The normalizer also changes effective decay, so the result characterizes the combined signal and strength choice.

6.4 AdamW Classification and Detection

ViT on CIFAR-100. The CIFAR ViT experiment tests whether the controller transfers to a Transformer under AdamW. We train a Vision Transformer (Dosovitskiy et al., 2021), ViT-Tiny with 16×16 patches, from scratch on CIFAR-100 using AdamW. The recipe resizes images to 224×224, uses batch size 512, learning rate $5 \cdot 10^{-4}$, 200 warm-up steps, and 90 epochs of cosine decay.

The selected EqWD setting $(\lambda_0, \beta, \alpha_\mu, \alpha_v) = (0.03, 0.75, 0.99, 0.9)$ reaches $53.980 \pm 0.663\%$ best-epoch accuracy. The reference fixed coefficient 0.05 reaches $54.007 \pm 0.432\%$. The paired difference is -0.027 ± 0.301 points. The searched fixed coefficient reaches $54.123 \pm 0.463\%$ best-epoch accuracy; all three means fall within a 0.143-point range. Appendix C.2 gives the screening procedure, initial and selected settings, and final-epoch results.

ViT on ImageNet-1K. The ImageNet ViT experiment tests whether the close CIFAR comparison persists with a longer training schedule. We use 300-epoch ImageNet-1K training with $\lambda_0 = 0.05$, $\beta = 0.5$, $\alpha_\mu = \alpha_v = 0.99$, and $\sigma_{\min} = 10^{-3}$ for EqWD. Effective batch sizes agree, but the EqWD seed-42 run uses a different microbatch/accumulation split; the appendix reports this execution difference and all individual scores.

Table 3 reports final-epoch results. EqWD reaches $73.739 \pm 0.060\%$ Top-1, compared with $73.718 \pm 0.113\%$ for fixed decay; the paired Top-1 difference is $+0.021 \pm 0.082$ points. Top-5 is slightly

lower, at $91.225 \pm 0.120\%$ versus $91.290 \pm 0.078\%$. The paired means are closely matched on this benchmark.

DINO on COCO2017. The COCO experiment tests the controller on detection with two backbones. EqWD raises overall AP from 48.9 to 49.5 with ResNet-50 and from 51.2 to 51.3 with Swin-T, corresponding to gains of 0.6 and 0.1 AP points.

Table 3 also exposes the scale and recall trade-offs: medium-object AP increases by 0.4 and 0.3 points, while small-object AP decreases by 0.6 and 0.4 points. AR_{100} increases by 0.4 points with ResNet-50 but decreases by 0.2 points with Swin-T. These comparisons describe the reported evaluations; statistical significance and uniform gains across scales remain untested. Table 4 gives the complete threshold and recall breakdown.

Table 3: Additional AdamW experiments. Panel (a) reports final-epoch ImageNet-1K validation accuracy (%). Panel (b) reports primary COCO2017 detection metrics on the 0–100 scale (↑); the full AP/AR breakdown is in Table 4.

(a) ViT-Tiny / ImageNet-1K (300 epochs)

Method	Top-1	Top-5
FixedWD	73.718 ± 0.113	91.290 ± 0.078
EqWD	73.739 ± 0.060	91.225 ± 0.120

(b) DINO / COCO2017

Backbone	Method	AP	AP_s	AP_m	AR_{100}
ResNet-50	FixedWD	48.9	31.6	52.1	72.3
	EqWD	49.5	31.0	52.5	72.7
Swin-T	FixedWD	51.2	34.2	53.8	74.1
	EqWD	51.3	33.8	54.1	73.9

Table 4: Full DINO detection breakdown on COCO2017 (0–100 scale, ↑). The evaluation reports both threshold-specific and size-specific metrics for the same four runs as Table 3.

(a) Bounding-box AP

Backbone	Method	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
ResNet-50	FixedWD	48.9	66.4	53.2	31.6	52.1	63.1
	EqWD	49.5	66.3	53.0	31.0	52.5	63.4
Swin-T	FixedWD	51.2	68.8	55.7	34.2	53.8	66.2
	EqWD	51.3	69.1	55.6	33.8	54.1	66.2

(b) Average recall

Backbone	Method	AR_{100}	AR_{300}	AR_{1000}	AR_s	AR_m	AR_l
ResNet-50	FixedWD	72.3	72.7	72.7	56.3	76.7	88.2
	EqWD	72.7	73.1	73.1	55.5	77.4	88.8
Swin-T	FixedWD	74.1	74.4	74.4	58.7	78.2	89.9
	EqWD	73.9	74.3	74.3	58.6	78.2	89.8

The full breakdown also shows an increase in large-object AP with ResNet-50 and an unchanged value with Swin-T. Recall improves most for medium objects under ResNet-50. Across the AdamW experiments, ViT remains close to fixed decay, whereas detection shows a gain in overall AP whose size depends on the backbone and which coexists with changes in scale-specific performance.

6.5 Controller Dynamics and Computational Cost

Logged coefficients remain below the analytical limits: the largest sampled multipliers are 11.981 on CIFAR/ResNet-20 and 10.449 on ImageNet/ResNet-50, against bounds of 12.180 and 11.000. The ImageNet/ResNet-50 phase means are 1.783, 1.778, 1.798, and 1.769, so additional decay persists throughout training. Figure 1 shows representative tensor traces with the tracked mean and modulation. The traces show how the moving mean follows changes in the ratio while the coefficient remains bounded; the phase means show that the additional decay persists across learning-rate reductions.

We measure computational cost on one NVIDIA L20X using synthetic GPU-resident minibatches. Five interleaved repeats per method each use 100 warm-up steps and 500 measured steps. The timed region covers forward/backward computation, manual decay, and the optimizer step; it excludes data loading and diagnostic logging. Table 5 reports the median step times.

The controller updates two recurrent scalar statistics per tensor. Norm reductions and scalar operations add computation; their relative cost is smaller in the ResNet-50 benchmark.

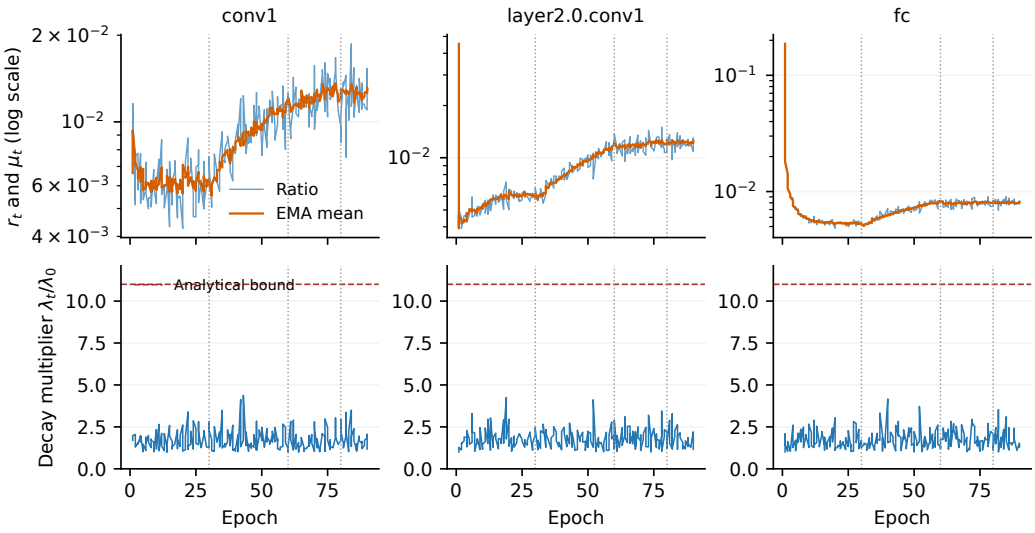

Figure 1: EqWD maintains a moving reference and persistent modulation. Representative seed-42 ImageNet tensors; the plotted ratio reconstructs the pre-decay detector input from logged states. Vertical lines mark learning-rate reductions. Appendix D explains sampling and reconstruction.

Table 5: Controlled median step time in milliseconds on one NVIDIA L20X. The overhead is relative to fixed decay. This synthetic-input benchmark measures compute cost; end-to-end training throughput is outside its scope.

Architecture	Fixed decay	EqWD	Overhead
ResNet-20	6.113	8.239	34.77%
ResNet-50	84.552	90.235	6.72%

7 Discussion

The gain and base coefficient jointly determine effective decay strength, so both enter the training recipe's regularization choice. The fixed-decay controls explain much of the reference CNN gain, while the ViT and DINO comparisons show how the outcome depends on architecture and metric. Exposure matching averages across sampled steps and eligible tensors; matching each tensor separately would help distinguish temporal adaptation from the allocation of decay across parameters. Appendices B.3 and B.4 provide the search grids and matching protocol for the current comparisons.

The theoretical results characterize the controller along finite trajectories and successful optimizer steps. For $0 < \alpha_\mu < 1$, vanishing ratio increments characterize recovery; the local convergence analysis covers a scalar quadratic with plain gradient descent. Momentum and preconditioning change the base update and require separate analysis. Finite precision also matters in implementation: small learning-rate–decay products can round the radial shrinkage to zero despite a positive nominal coefficient. Appendix B specifies the update ordering and numerical conventions used in the experiments.

8 Conclusion

EqWD uses a moving gradient-to-weight ratio reference to control decay with a bounded coefficient. Its EMA states provide a change-response formula, recovery characterization, and bounds on instantaneous and cumulative modulation. Fixed-decay controls explain much of the CNN gain; ViT remains close to fixed decay, while DINO improves overall AP with backbone-dependent trade-offs. These results define the controller's scope and evidence limits.

AI USAGE STATEMENT

Large language models assisted with experimental code writing, manuscript restructuring and editing, proof checking, and analysis of existing experiment records. They also provided feedback on methodology and interpretation. The numerical tables derive from experiment records and supplied run summaries; the appendix states the DINO evaluation scope. The authors remain responsible for the final claims and submission.

REFERENCES

Andy Brock, Soham De, Samuel L. Smith, and Karen Simonyan. High-performance large-scale image recognition without normalization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 1059–1071, 2021. URL https://arxiv.org/abs/2102.06171.

Lizhang Chen, Jonathan Li, Kaizhao Liang, Baiyu Su, Cong Xie, Nuo Wang Pierse, Chen Liang, Ni Lao, and Qiang Liu. Cautious weight decay, 2026. URL https://arxiv.org/abs/2510.12402v2. arXiv version 2, revised February 2026.

Aaron Defazio. Why gradients rapidly increase near the end of training, 2025. URL https://arxiv.org/abs/2506.02285.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.11929.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.01412.

Jörg K. H. Franke, Michael Hefenbrock, Gregor Koehler, and Frank Hutter. Improving deep learning optimization through constrained parameter regularization. In *Advances in Neural Information Processing Systems*, volume 37, pp. 8984–9025, 2024. URL https://arxiv.org/abs/2311.09058.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: Training ImageNet in 1 hour, 2017. URL https://arxiv.org/abs/1706.02677.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016. URL https://arxiv.org/abs/1512.03385.

Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors, 2012. URL https://arxiv.org/abs/1207.0580.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1609.04836.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

Atli Kosson, Bettina Messmer, and Martin Jaggi. Rotational equilibrium: How weight decay balances learning across neural networks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 25333–25369, 2024. URL https://arxiv.org/abs/2305.17212.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common objects in context. In *Computer Vision – ECCV 2014*, 2014. URL https://arxiv.org/abs/1405.0312.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. URL https://arxiv.org/abs/2103.14030.

Ilya Loshchilov. Weight norm control, 2023. URL https://arxiv.org/abs/2311.11446.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1608.03983.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10347–10357, 2021. URL https://arxiv.org/abs/2012.12877.

Ashia C. Wilson, Rebecca Roelofs, Mitchell Stern, Nathan Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://arxiv.org/abs/1705.08292.

Zeke Xie, Zhiqiang Xu, Jingzhao Zhang, Issei Sato, and Masashi Sugiyama. On the overlooked pitfalls of weight decay and how to mitigate them: A gradient-norm perspective. In *Advances in Neural Information Processing Systems*, volume 36, pp. 1208–1228, 2023. URL https://arxiv.org/abs/2011.11152.

Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks, 2017. URL https://arxiv.org/abs/1708.03888.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ming shuan Ni, and Heung-Yeung Shum. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2203.03605.

A PROOFS AND SCOPE OF THE GUARANTEES

We analyze one eligible parameter tensor and suppress its tensor index. The arguments assume finite real-valued observations, $\alpha_\mu, \alpha_v \in [0, 1)$, $\sigma_{\min} > 0$, $\varepsilon > 0$, and $\lambda_0, \beta \geq 0$. We index the first observation by $t = 0$, with $\mu_0 = r_0$ and $v_0 = 0$. Thus $e_0 = q_0 = 0$ and $\lambda_0(1 + \beta q_0) = \lambda_0$. The subscript on the constant λ_0 denotes base decay throughout. Write $a = \alpha_\mu$, $b = \alpha_v$, $s = \sigma_{\min}$, and $d = s + \varepsilon$.

A.1 COEFFICIENT BOUND

Proof of Proposition 1. Nonnegative initialization and the residual second-moment recursion give

$$v_t \geq (1 - \alpha_v)e_t^2 \geq 0 \qquad (t \geq 1).$$

If $e_t = 0$, then $q_t = 0$ because $D_t > 0$. If $e_t \neq 0$,

$$q_t = \frac{|e_t|}{D_t} \leq \frac{|e_t|}{\sqrt{v_t}} \leq \frac{1}{\sqrt{1 - \alpha_v}}.$$

The initialized step also satisfies this bound. Multiplication by $\lambda_0\beta$ and addition of λ_0 give the coefficient interval in the proposition. If $0 \leq \gamma_t\lambda_{\max} \leq 1$, then $0 \leq 1 - \gamma_t\lambda_t \leq 1$ and hence

$$\|(1 - \gamma_t\lambda_t)w_t\| = (1 - \gamma_t\lambda_t)\|w_t\| \leq \|w_t\|.$$

The statement concerns the norm after the decay branch at the realized coefficient. It gives no Lipschitz bound between two state-dependent trajectories. □

The gain bound requires no upper bound on r_t or e_t because the update includes e_t^2 in v_t before computing q_t. Using v_{t-1} in the denominator would remove this guarantee: an arbitrarily large current residual could follow a zero residual state.

For $\alpha_\mu > 0$, the constant is sharp as a supremum. Start with $\mu_0 = 0$, $v_0 = 0$, and choose $r_1 = R > 0$. Then $e_1 = \alpha_\mu R$ and $v_1 = (1 - \alpha_v)\alpha_\mu^2 R^2$, so $q_1 \to (1 - \alpha_v)^{-1/2}$ as $R \to \infty$. When $\alpha_\mu = 0$, the updated mean equals the observation and $q_t = 0$ at every step. The choice $\alpha_v = 1$ lies outside the assumptions and removes the current-residual contribution.

A.2 RATIO-CHANGE FILTER

Proof of Proposition 2. The mean update gives

$$e_t = r_t - a\mu_{t-1} - (1 - a)r_t = a(r_t - \mu_{t-1}) = a(\Delta r_t + e_{t-1}).$$

Unrolling from $e_0 = 0$ proves Equation (8). If $a = 0$, all residuals and modulation signals equal zero. Suppose henceforth that $0 < a < 1$.

If $\Delta r_t \to 0$, then for any $\delta > 0$ there exists T such that $|\Delta r_{T+j}| \leq \delta$ for all $j \geq 1$. Unrolling the residual recursion gives

$$|e_{T+k}| \leq a^k|e_T| + a\sum_{j=1}^{k} a^{k-j}|\Delta r_{T+j}| \leq a^k|e_T| + \frac{a(1 - a^k)}{1 - a}\delta.$$

Taking the upper limit and then $\delta \to 0$ gives $e_t \to 0$. Since $D_t \geq d > 0$, we obtain $q_t \leq |e_t|/d \to 0$. Conversely, $e_t \to 0$ implies $\Delta r_t = e_t/a - e_{t-1} \to 0$.

It remains to show that $q_t \to 0$ implies $e_t \to 0$ without assuming an upper bound on v_t. Choose $\delta \in (0, 1)$ and T such that $q_t \leq \delta$ for every $t \geq T$. Whenever $v_t \geq v_{t-1}$, the scale recursion implies $v_t \leq e_t^2$. At such a step,

$$\sqrt{v_t} \leq |e_t| = q_t D_t \leq \delta(\sqrt{v_t} + d), \qquad v_t \leq \left(\frac{\delta d}{1 - \delta}\right)^2.$$

On decreasing steps, $v_t \leq v_{t-1}$. Induction therefore bounds the entire tail by $\max\{v_{T-1}, (\delta d/(1 - \delta))^2\}$. Thus D_t remains bounded and $|e_t| = q_t D_t \to 0$. The three limits are therefore equivalent. The coefficient equivalence follows from $\lambda_t - \lambda_0 = \lambda_0\beta q_t$ when $\lambda_0\beta > 0$. If this product is zero, decay equals λ_0 regardless of q_t. □

The tail bound in the proof gives a finite-time response to slowly varying ratios. Starting from $e_0 = 0$ and imposing $|\Delta r_t| \le L$ yields Equation (10). For a constant tail $r_{T+k} = R$ for $k \ge 1$,

$$q_{T+k} \le \frac{a^k |R - \mu_T|}{d}.$$

If $r_t \to R$, the mean also converges to R. Vanishing residuals imply $v_t \to 0$ by the same geometric-tail argument. The stronger recovery condition permits unbounded inputs such as $r_t = \log(t+1)$, whose increments vanish.

The positive denominator lower bound matters for recovery. To see this, temporarily remove both stabilizers and set $D_t = \sqrt{v_t}$. For a constant tail with nonzero initial mean error and $a^2 > b$, $e_{T+k} = a^k(R - \mu_T)$ and

$$\frac{v_{T+k}}{e_{T+k}^2} \longrightarrow \frac{1-b}{1-b/a^2}, \qquad q_{T+k} \longrightarrow \sqrt{\frac{1-b/a^2}{1-b}} > 0.$$

Vanishing residuals alone can leave standardized deviations positive when the denominator vanishes at the same rate. The implemented positive floor prevents this failure.

A.3 Potential and energy bound

Proof of Theorem 1. The function H_s in Equation (11) is continuously differentiable and concave on $[0, \infty)$, with

$$H_s(0) = 0, \qquad H_s'(v) = \frac{1}{\max(v, s^2)}.$$

Its tangent inequality at v_t gives

$$(v_t - v_{t-1}) H_s'(v_t) \le H_s(v_t) - H_s(v_{t-1}).$$

The inequality holds for both increasing and decreasing scale. Solving the scale recursion for e_t^2 yields

$$e_t^2 = v_t + \frac{b}{1-b}(v_t - v_{t-1}).$$

Since $D_t^2 \ge \max(v_t, s^2)$, the previous two relations give

$$\begin{aligned} q_t^2 &\le e_t^2 H_s'(v_t) \\ &= \min\{v_t/s^2, 1\} + \frac{b}{1-b}(v_t - v_{t-1}) H_s'(v_t) \\ &\le \min\{v_t/s^2, 1\} + \frac{b}{1-b}\big[H_s(v_t) - H_s(v_{t-1})\big]. \end{aligned}$$

Summation telescopes from $v_0 = 0$, proving Equation (12). Cauchy–Schwarz gives $T^{-1}\sum_t q_t \le (T^{-1}\sum_t q_t^2)^{1/2}$. Substituting $\lambda_t = \lambda_0(1 + \beta q_t)$ proves Equation (13). For $b = 0$, the potential term vanishes and $q_t \le 1$ already gives the result. □

The potential grows only logarithmically above the scale floor. If $H_s(v_T) = o(T)$, then

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} q_t \le 1, \qquad \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \lambda_t \le \lambda_0(1+\beta).$$

Bounded or polynomially growing v_T satisfies this condition. The finite-time theorem requires neither condition. The first bound in Equation (12) also accounts for steps below the floor through $\min\{v_t/s^2, 1\}$.

A.4 Learning-rate-weighted exposure

The accumulated additional decay exposure is

$$\mathcal{E}_T = \sum_{t=1}^{T} \gamma_t(\lambda_t - \lambda_0).$$

The quantity weights the coefficient by the learning rate used in the parameter update. For arbitrary $\gamma_t \geq 0$, Equation (8) gives

$$\mathcal{E}_T \leq \frac{\lambda_0 \beta a}{d} \sum_{k=1}^{T} |\Delta r_k| \sum_{t=k}^{T} \gamma_t a^{t-k}. \tag{14}$$

The denominator floor gives $q_t \leq d^{-1} \sum_{k=1}^{t} a^{t-k+1} |\Delta r_k|$; multiply by $\gamma_t \lambda_0 \beta$ and interchange the finite sums.

Suppose now that $\gamma_1 \geq \cdots \geq \gamma_T \geq 0$. Let $\Gamma_T = \sum_{t=1}^{T} \gamma_t$ and $V_T = \max_{0 \leq t \leq T} v_t$. Two complementary exposure bounds follow:

$$\mathcal{E}_T \leq \lambda_0 \beta \min \left\{ \frac{a}{(1-a)d} \sum_{k=1}^{T} \gamma_k |\Delta r_k|, \sqrt{\Gamma_T \left[\Gamma_T + \frac{b}{1-b} \gamma_1 H_s(V_T) \right]} \right\}. \tag{15}$$

For the first bound, use $\gamma_t \leq \gamma_k$ when $t \geq k$ in Equation (14) and sum a geometric series. For the second, multiply the one-step potential inequality by γ_t. Summation by parts gives

$$\begin{aligned} \sum_{t=1}^{T} \gamma_t \big[H_s(v_t) - H_s(v_{t-1}) \big] &= \gamma_T H_s(v_T) + \sum_{t=1}^{T-1} (\gamma_t - \gamma_{t+1}) H_s(v_t) \\ &\leq \gamma_1 H_s(V_T). \end{aligned}$$

Hence $\sum_t \gamma_t q_t^2 \leq \Gamma_T + \frac{b}{1-b} \gamma_1 H_s(V_T)$. Weighted Cauchy–Schwarz proves the second bound, including $\Gamma_T = 0$, where both sides vanish. For a schedule with warmup, Equation (14) still applies to the complete schedule. The monotone-schedule bound applies directly from zero initialization only when the full sequence of learning rates is nonincreasing. The bounds apply along a realized trajectory and make no cross-optimizer loss comparison.

A.5 Radial decay and bounded updates

For the same current state and base-optimizer direction u_t, the difference between EqWD and fixed decay with coefficient λ_0 is

$$\delta_t = -\gamma_t \lambda_0 \beta q_t w_t.$$

If $w_t \neq 0$, its projection onto the tangent space orthogonal to w_t satisfies

$$\left(I - \frac{w_t w_t^\top}{\|w_t\|^2} \right) \delta_t = 0.$$

If $w_t = 0$, then $\delta_t = 0$. The identity describes the adaptive branch at a shared state. Subsequent gradients and task directions can differ between complete training trajectories.

An additional bounded-input assumption gives a bound on the full update. Suppose $\lambda_0 > 0$, $\|u_t\| \leq U$, and $0 \leq \gamma_t \lambda_{\max} \leq 1$ at every step. Then

$$\begin{aligned} \|w_{t+1}\| &\leq (1 - \gamma_t \lambda_t) \|w_t\| + \gamma_t U \\ &\leq (1 - \gamma_t \lambda_0) \|w_t\| + \gamma_t U. \end{aligned}$$

Let $B = \max\{\|w_0\|, U/\lambda_0\}$. If $\|w_t\| \leq B$, the last expression is at most $B - \gamma_t \lambda_0 B + \gamma_t U \leq B$. Induction therefore gives $\|w_t\| \leq B$ for all t. The argument applies to each eligible tensor and requires a bound on its base-optimizer direction. For SGD momentum $b_t = m b_{t-1} + g_t$, $b_{-1} = 0$, and $0 \leq m < 1$, a gradient bound $\|g_t\| \leq G$ gives $\|b_t\| \leq G/(1-m)$. One may then take $U = G/(1-m)$. The coefficient bound supplies no gradient assumption.

Bounded coefficients and a norm-reducing decay branch also allow divergence when the gradient step is too large. Consider the scalar loss $f(w) = w^2/2$, plain gradient descent, $\gamma = 3$, $\lambda_0 = 0.01$, $\beta = 1$, and $\alpha_v = 0.9$. Here $\gamma \lambda_{\max} < 0.125$, yet

$$w_{t+1} = [1 - 3(1 + \lambda_t)] w_t, \qquad |w_{t+1}| \geq 2.03 |w_t|.$$

Every nonzero initial parameter therefore diverges in magnitude. The coefficient bound describes the decay branch; the full optimizer dynamics also depend on the base update.

A.6 Alternating ratio observations

Consider $r_t = R + A(-1)^t$ with $R > A > 0$ and write $a = \alpha_\mu$. The mean recursion approaches the periodic solution

$$\mu_t = R + \frac{1-a}{1+a} A(-1)^t + o(1).$$

Substitution verifies the recursion, and the difference from any finite initial mean decays as a^t. It follows that

$$e_t = \frac{2aA}{1+a}(-1)^t + o(1).$$

Writing $E = 2aA/(1+a)$, the second-moment recursion gives $v_t \to E^2$. Consequently,

$$q_t \longrightarrow \frac{E}{\max(E, \sigma_{\min}) + \varepsilon}.$$

For $a > 0$ and E large relative to the stabilizers, this limit is close to one and the effective decay approaches a value close to $\lambda_0(1+\beta)$. For $a = 0$, the residual and modulation remain zero. A uniformly random initial phase makes this alternating process strictly stationary with mean R, while preserving the calculation. Stationarity of the ratio distribution is therefore insufficient for recovery of base decay. The increments alternate with magnitude $2A$, as Proposition 2 predicts. Taking long-time averages and then increasing A makes the mean modulation approach one. The asymptotic constant in Theorem 1 is sharp over these input sequences when $a > 0$.

A.7 Scalar quadratic feedback loop

A scalar quadratic permits analysis of the actual feedback loop, including the moving mean and residual scale. Consider $f(w) = \kappa(w-c)^2/2$ with $\kappa, c, \lambda_0 > 0$. Use the EqWD recurrence with $u_t = f'(w_t)$ and a constant learning rate $0 < \eta \le 1/(\kappa + \lambda_0)$. The ratio remains the implemented $r(w) = |\kappa(w-c)|/(|w| + \varepsilon)$. The joint fixed point is

$$w_* = \frac{\kappa c}{\kappa + \lambda_0}, \qquad r_* = \frac{\lambda_0 w_*}{w_* + \varepsilon}, \qquad \mu_* = r_*, \qquad v_* = 0.$$

Define

$$k = 1 - \eta(\kappa + \lambda_0) \in [0, 1), \qquad h = \frac{\eta\lambda_0\beta w_*}{d} \frac{\kappa(c+\varepsilon)}{(w_* + \varepsilon)^2}.$$

Proposition 3 (A sufficient local feedback condition)**.** *Under these assumptions and the controller parameter ranges in §4, the joint fixed point is locally exponentially stable if*

$$2ah < 1 - a. \tag{16}$$

Sufficiently close initialized trajectories satisfy $w_t \to w_*$, $\mu_t \to r_*$, $v_t \to 0$, $q_t \to 0$, and $\lambda_t \to \lambda_0$ at geometric rates.

Proof. Use the pre-update state $(w_t, \mu_{t-1}, v_{t-1})$. Setting $\mu_{-1} = r(w_0)$ and $v_{-1} = 0$ reproduces the first-observation initialization in §4. Since $0 < w_* < c$, the ratio is smooth and strictly decreasing in a neighborhood of w_*:

$$r(w) = \frac{\kappa(c-w)}{w+\varepsilon}, \qquad r'(w) = -\frac{\kappa(c+\varepsilon)}{(w+\varepsilon)^2}.$$

Thus $z_t = r(w_t) - r_*$ is a smooth local coordinate. Write $m_t = \mu_{t-1} - r_*$ and $\xi_t = z_t - m_t$. The mean update gives exactly $m_{t+1} = z_t - a\xi_t$ and $e_t = a\xi_t$. Whenever $0 \le v_{t-1} < s^2$ and $|a\xi_t| < s$, the scale update gives $v_t < s^2$ and hence $D_t = d$. The parameter update then reads

$$w_{t+1} - w_* = k(w_t - w_*) - \frac{\eta\lambda_0\beta a}{d}|\xi_t| w_t.$$

Taylor expansion of r and its smooth local inverse gives

$$\begin{aligned} z_{t+1} &= kz_t + ha|\xi_t| + R_t, \\ \xi_{t+1} &= (k-1)z_t + a\xi_t + ha|\xi_t| + R_t, \end{aligned}$$

where $|R_t| \le C(|z_t| + |\xi_t|)^2$ in a fixed neighborhood. The bound holds on both signs of ξ_t because the displayed absolute value contains the entire first-order nonsmooth term.

For $0 < a < 1$, Equation (16) implies $1 - a(1+h) > ah \ge 0$. Choose

$$\frac{ah}{1-a(1+h)} < p < 1, \qquad N_t = |z_t| + p|\xi_t|.$$

Using $0 \le k < 1$ in the preceding recursions yields

$$N_{t+1} \le \rho N_t + C_1 N_t^2, \qquad \rho = \max\left\{k + p(1-k),\ a(1+h) + \frac{ah}{p}\right\} < 1.$$

Choose $\delta > 0$ small enough that the Taylor bound holds for $N_t \le \delta$, $a\delta/p < s$, and $C_1\delta \le (1-\rho)/2$. If $N_0 \le \delta$ and $0 \le v_{-1} < s^2$, induction gives $N_t \le \tau^t N_0$ with $\tau = (1+\rho)/2 < 1$, while $v_t < s^2$ keeps the floor active. Shrinking δ also keeps the parameter update inside the coordinate neighborhood.

Finally,

$$v_t = b^{t+1} v_{-1} + (1-b)a^2 \sum_{j=0}^{t} b^{t-j} \xi_j^2.$$

For any $\theta \in (\max\{b, \tau^2\}, 1)$, the geometric convolution is at most $C_2\theta^t(v_{-1} + N_0^2)$ for a finite C_2. The full state converges exponentially, including the cases $b = 0$ and $b = \tau^2$. When $a = 0$, $e_t = q_t = 0$ exactly, the parameter error contracts by k, and the scale contracts by b. The case $a = 0$ follows directly. Standard initialization gives $\xi_0 = 0$ and $N_0 = |r(w_0) - r_*|$, which is sufficiently small whenever w_0 is sufficiently close to w_*. □

Condition (16) is sufficient; necessity remains open. The condition links the gain β, mean memory a, and denominator floor d through h. The scale memory b affects the transient but drops out of this local condition because the floor is active near the fixed point. The positive term $ha|\xi_t|$ in the ratio recurrence retains the feedback direction highlighted by the ratio geometry. Local convergence follows when the base update and mean tracking dominate that term under Equation (16). The limit w_* minimizes $f(w) + \lambda_0 w^2/2$. Extending this calculation to other optimizers and model classes changes the base-update terms.

B Experimental setup and evaluation protocols

B.1 Datasets and training configurations

The CIFAR-100 experiments use the official 50,000-image training split and 10,000-image test split. ResNet-20 has 278,324 parameters. Its training pipeline applies a random 32×32 crop with padding four, horizontal flipping, and channel normalization. Evaluation applies normalization without augmentation.

ImageNet-1K training uses 1,281,167 images and evaluates on 50,000 validation images. The loader streams the data from 294 training and 14 validation parquet files. Training applies random resized cropping to 224×224, horizontal flipping, color jitter $(0.4, 0.4, 0.4)$, and ImageNet channel normalization. Evaluation resizes the short side to 256 and takes a 224×224 center crop. ResNet-50 has 25,557,032 parameters.

The ViT configuration uses `vit_tiny_patch16_224` from `timm`, with 5,543,716 parameters and no pretrained weights. The small architecture follows the ViT/DeiT family (Dosovitskiy et al., 2021; Touvron et al., 2021). For CIFAR training, it uses random resized cropping to 224 with scale $(0.8, 1.0)$, bicubic interpolation, horizontal flipping, and normalization. Evaluation resizes CIFAR images to 224 with bicubic interpolation. These CIFAR ViT experiments use cross-entropy loss with zero label smoothing; dropout and stochastic depth rates are zero. Each run seeds Python, NumPy, PyTorch, and CUDA with its listed seed; the source enables deterministic cuDNN behavior and disables its benchmark mode.

For SGD, momentum is 0.9, dampening is zero, and Nesterov momentum is off. AdamW uses $(\beta_1, \beta_2) = (0.9, 0.999)$ and optimizer epsilon 10^{-8}. All methods set the optimizer's native decay

to zero and apply decoupled decay manually before the optimizer step. The CIFAR and ImageNet classification runs use automatic mixed precision (AMP).

The loader drops incomplete training batches, and CIFAR evaluation uses twice the training batch size. ResNet-50/ImageNet validation retains batch size 256; ViT/ImageNet validation uses the run's microbatch size. CIFAR ResNet experiments use four workers. ImageNet and the initial ViT experiments use eight workers, while the later ViT tuning experiments use sixteen. Worker count can affect stochastic augmentation order, so matching seed identifiers across these ViT stages can still produce different data streams.

Supplementary ImageNet ViT and DINO/COCO protocols. The additional ImageNet classification and COCO detection comparisons use the supplied run summaries and reported configurations. For the ViT-Tiny/ImageNet-1K comparison, we use the same `vit_tiny_patch16_224` architecture without pretrained weights. Training uses 224×224 inputs, AdamW, 300 epochs, and AMP. The effective batch size is 4096, the initial learning rate is 0.004, and five epochs of linear warm-up from 10^{-6} precede cosine decay to 10^{-5}. The recipe uses base decay 0.05, dropout 0, stochastic-depth rate 0.1, and label smoothing 0.1. EqWD additionally uses $\beta = 0.5$, $\alpha_\mu = \alpha_v = 0.99$, and $\sigma_{\min} = 10^{-3}$. All three FixedWD runs use microbatch 512 with eight accumulation steps. The EqWD seed-42 run uses microbatch 1024 with four accumulation steps; its seed-123 and seed-456 runs use 512 with eight steps. Every run therefore has the same effective batch size, while seed 42 uses a different microbatch execution.

The supplied detection configuration uses four-scale DINO on COCO2017 `train2017` and evaluates on `val2017` after every epoch. The backbones are ImageNet-pretrained ResNet-50 and Swin-T; the Swin-T configuration uses dimensions $(96, 192, 384, 768)$, depths $(2, 2, 6, 2)$, window size 7, and stochastic-depth rate 0.2. The standard DINO pipeline applies random horizontal flipping and multi-scale resizing, with a random crop branch. Each run uses seed 42, 12 epochs, microbatch 8, two gradient accumulation steps, and effective batch size 16 on one GPU. AdamW uses $(\beta_1, \beta_2) = (0.9, 0.999)$, epsilon 10^{-8}, learning rate 10^{-4}, and a 0.1 backbone learning-rate multiplier. The learning rate drops tenfold at epoch 11. Both methods use nominal base decay 10^{-4}; EqWD uses $\beta = 0.5$, $\alpha_\mu = \alpha_v = 0.99$, and $\sigma_{\min} = 10^{-3}$. The launcher clips gradients to global norm 0.1 after accumulation, disables AMP, and uses four/two workers for the training/validation loaders. The checkpoint hook saves the best validation box AP. The supplied launcher and generated configuration define these settings; the metric summary reports the resulting comparison.

Table 6: Reference FixedWD and EqWD settings for the main experiments. Dashes indicate controller parameters that are inapplicable to FixedWD. The table lists the reference fixed coefficient; Appendix B.3 specifies searched and exposure-matched controls, and Appendix C.2 specifies the selected CIFAR ViT entries.

Setting	Method	λ_0	β	α_μ	α_v	$\sigma_{\min}$	Epochs
CIFAR / ResNet-20	FixedWD	$5 \cdot 10^{-4}$	–	–	–	–	200
	EqWD	$5 \cdot 10^{-4}$	5	0.99	0.80	10^{-4}	200
ImageNet / ResNet-50	FixedWD	10^{-4}	–	–	–	–	90
	EqWD	10^{-4}	1	0.90	0.99	10^{-4}	90
CIFAR / ViT-Tiny	FixedWD	0.05	–	–	–	–	90
	EqWD	0.03	0.75	0.99	0.90	10^{-3}	90
ImageNet / ViT-Tiny	FixedWD	0.05	–	–	–	–	300
	EqWD	0.05	0.5	0.99	0.99	10^{-3}	300
COCO / DINO (both backbones)	FixedWD	10^{-4}	–	–	–	–	12
	EqWD	10^{-4}	0.5	0.99	0.99	10^{-3}	12

B.2 Decay implementation

The runs set `layer_aware=false`; this disables an optional layer-type adjustment while retaining tensor-specific EMA states. The uniform rule still tracks each eligible tensor separately, and the no-decay filter excludes tensors of dimension at most one, biases, names identifying normalization parameters, position embeddings, class tokens, and distillation tokens. Tensors without a gradient

Table 7: Saved result records by experiment stage. Counts include screening and follow-up evaluation. Each main comparison uses three seeds.

Study	Runs	Epochs per run
CIFAR equal-count screen	24	100
CIFAR selected configurations	6	200
CIFAR exposure matching	3	200
CIFAR seven component variants	21	200
ImageNet preflight	1	1
ImageNet reference and EqWD	6	90
ImageNet exposure matching	3	90
Initial ViT comparison	6	90
ViT fixed/EqWD screen	16	90
Additional ViT EqWD screen	8	90
ViT confirmation (seeds 123, 456)	8	90
CIFAR recipe variants	24	200
Total completed records	126	

receive no manual decay on that step. The evidence includes source snapshots for the exact filter and optimizer ordering.

In mixed-precision training, the source unscales gradients before computing EqWD statistics. For a tensor whose gradient norm is nonfinite, the wrapper falls back to its base decay and leaves that tensor's EMA states unchanged. Manual decay occurs before `GradScaler.step`. An overflow can apply decay before the base optimizer skips its update. The saved summaries exclude skipped steps.

B.3 Hyperparameter and checkpoint selection

Checkpoint and hyperparameter choices follow the benchmark evaluation protocol described here. Both best-epoch and final-epoch scores use that protocol. CIFAR evaluation and screening use its official test split, and ImageNet uses its validation split; the experiments add no separate held-out split.

The CIFAR screen uses seed 42 for 100 epochs with twelve trials per method. The fixed-decay grid is

$$\{0, 10^{-5}, 3\cdot10^{-5}, 10^{-4}, 2\cdot10^{-4}, 3\cdot10^{-4}, 5\cdot10^{-4}, 7\cdot10^{-4}, 10^{-3}, 2\cdot10^{-3}, 5\cdot10^{-3}, 10^{-2}\}.$$

For EqWD, the reference tuple $(\lambda_0, \beta, \alpha_\mu, \alpha_v)$ is $(5\cdot10^{-4}, 1, 0.99, 0.8)$. The twelve settings comprise this tuple, three alternatives for $\lambda_0 \in \{10^{-4}, 3\cdot10^{-4}, 10^{-3}\}$, three for $\beta \in \{0.5, 2, 5\}$, two for $\alpha_\mu \in \{0.9, 0.95\}$, and three for $\alpha_v \in \{0.9, 0.95, 0.99\}$. Each alternative changes only the indicated entry. The screen selects fixed decay 0.01 and the EqWD setting in Table 6. The trial counts match, but the grids span different parameter spaces. The results provide no evidence about global optima.

The ImageNet study uses the specified EqWD setting, the reference fixed coefficient, and the estimated exposure-matched control. It includes no fixed-coefficient sweep.

The total includes a one-epoch preflight. The selected CIFAR EqWD and full-EqWD rows share histories for the three listed seeds, so the table counts one observation per seed. The run index records training flags and hashes for the saved result files.

B.4 Exposure-matched fixed-decay controls

Let n_l be the number of parameters in eligible tensor l and $\mathcal{T}_s$ the diagnostic steps for seed s. The matched coefficient approximates

$$\widehat{\lambda}_s = \frac{\sum_{t\in\mathcal{T}_s} \gamma_t \sum_l n_l \lambda_t^l}{\sum_{t\in\mathcal{T}_s} \gamma_t \sum_l n_l}, \qquad \widehat{\lambda} = \frac{1}{3}\sum_s \widehat{\lambda}_s. \tag{17}$$

The derivation reconstructs the schedule at sampled steps, using cosine learning-rate weights on CIFAR and step-schedule weights on ImageNet. The saved derivations give 0.0024862354008556077

Table 8: Best / final top-1 accuracy (%) for each seed. The accompanying generation script obtains these values directly from the archived result JSON files.

Configuration	Seed 42	Seed 123	Seed 456
CIFAR reference fixed	65.190 / 64.950	65.010 / 64.860	65.060 / 64.780
CIFAR searched fixed	68.990 / 68.850	69.020 / 68.530	69.530 / 69.320
CIFAR matched fixed	67.770 / 67.570	67.740 / 67.570	67.810 / 67.740
CIFAR EqWD	67.240 / 66.910	67.490 / 67.290	68.150 / 67.970
ImageNet reference fixed	72.848 / 72.804	72.960 / 72.922	72.528 / 72.454
ImageNet matched fixed	72.624 / 72.618	74.198 / 74.070	73.896 / 73.790
ImageNet EqWD	73.706 / 73.698	73.934 / 73.816	73.844 / 73.836
ViT reference fixed	54.300 / 54.190	53.510 / 53.320	54.210 / 54.200
ViT initial EqWD	53.720 / 53.630	53.660 / 53.470	54.220 / 54.010
ViT selected EqWD	54.180 / 54.030	53.240 / 53.030	54.520 / 54.450
ViT searched fixed 0.03	54.040 / 53.950	53.570 / 53.400	54.550 / 54.330
ViT searched fixed 0.04	54.530 / 54.410	53.620 / 53.470	54.220 / 54.190

Table 9: CIFAR-100/ResNet-20 recipe changes. Mean $\pm$ sample standard deviation over three seeds. All rows use configurations selected in the original cosine recipe.

Recipe	Method	Best epoch	Final epoch
Batch 64	Fixed, 0.01	69.053 ± 0.435	68.813 ± 0.333
Batch 64	EqWD	68.767 ± 0.385	68.683 ± 0.310
Batch 256	Fixed, 0.01	68.253 ± 0.367	68.157 ± 0.344
Batch 256	EqWD	66.417 ± 0.621	66.287 ± 0.675
Five-epoch warm-up	Fixed, 0.01	69.037 ± 0.539	68.930 ± 0.572
Five-epoch warm-up	EqWD	68.000 ± 0.545	67.780 ± 0.753
Step schedule	Fixed, 0.01	68.823 ± 0.133	67.757 ± 0.204
Step schedule	EqWD	68.677 ± 0.251	67.950 ± 0.416

and 0.0001790985475050136, respectively. The launched controls round these to 0.002486235401 and 0.0001790985475. One coefficient serves all tensors and seeds in each control.

For small $\gamma_t \lambda_t^l$, cumulative linear exposure approximates the negative log of the product $\prod_t (1-\gamma_t \lambda_t^l)$. Equation (17) averages over tensors and sampled steps, so exact cumulative shrinkage can differ across tensors. The diagnostics also omit unsampled updates. This control tests whether comparable global strength explains the observed gain; it cannot isolate temporal or tensor allocation effects.

B.5 Results by seed

For each matched seed pair, we subtract the comparator's accuracy from EqWD's accuracy and compute the mean and sample standard deviation of those three differences. These summaries describe run variability across three seeds.

C Additional experiments

C.1 Transfer across training recipes

Table 9 compares the selected fixed coefficient 0.01 and the selected EqWD configuration under four recipe changes. Each comparison uses 200 epochs and three seeds. Batch-size changes retain cosine decay; the warm-up setting adds five epochs of linear warm-up; the step setting reduces the learning rate tenfold at epochs 100 and 150. The runs transfer the selected coefficients without a new search for each recipe. Fixed decay has the higher mean best-epoch accuracy in all four. EqWD has the higher final-epoch mean with the step schedule.

Table 10: ViT-Tiny/AdamW on CIFAR-100 with reference and selected settings. All runs use 90 epochs. The aggregate for each tuned configuration includes its seed-42 screening run.

Configuration	Best epoch	Final epoch
reference fixed	54.007 ± 0.432	53.903 ± 0.505
initial EqWD	53.867 ± 0.307	53.703 ± 0.277
selected EqWD	53.980 ± 0.663	53.837 ± 0.729
searched fixed 0.03	54.053 ± 0.490	53.893 ± 0.468
searched fixed 0.04	54.123 ± 0.463	54.023 ± 0.492

C.2 ViT-Tiny configuration selection

The initial ViT comparison uses fixed decay 0.05 and EqWD $(\lambda_0, \beta, \alpha_\mu, \alpha_v) = (0.05, 0.5, 0.99, 0.99)$, with floor 10^{-3}. The subsequent seed-42 screen tests fixed coefficients $\{0, 0.005, 0.01, 0.02, 0.03, 0.04, 0.06, 0.08\}$. It also tests eight EqWD settings, holding $(\alpha_\mu, \alpha_v) = (0.99, 0.99)$:

$$(\lambda_0, \beta) \in \{(0.02, 0.5), (0.025, 0.5), (0.0275, 1), (0.03, 0.5), \\ (0.035, 0.5), (0.04, 0.25), (0.04, 0.5), (0.045, 0.5)\}.$$

An additional eight-setting EqWD screen holds $\lambda_0 = 0.03$ and varies β and the EMA rates. Six settings combine $\beta \in \{0.25, 0.5, 0.75\}$ with $(\alpha_\mu, \alpha_v) \in \{(0.9, 0.99), (0.99, 0.9)\}$; two use $(\beta, \alpha_\mu, \alpha_v) = (0.5, 0.999, 0.99)$ and $(0.5, 0.99, 0.999)$.

Confirmation adds seeds 123 and 456 for fixed coefficients 0.03 and 0.04 and two EqWD configurations. The seed-42 screen selects candidates before confirmation with seeds 123 and 456. The three-seed aggregates reuse the screening result for seed 42. Table 10 includes both fixed coefficients and the initial and selected EqWD settings. The additional EqWD confirmation with $(0.03, 0.5, 0.99, 0.99)$ reaches $53.590 \pm 0.576\%$ best and $53.517 \pm 0.548\%$ final accuracy. Among the evaluated settings, this configuration has the highest available three-seed EqWD mean.

C.3 ViT-Tiny on ImageNet-1K

Table 11 lists the six supplied run rows. The main table uses final-epoch Top-1 and Top-5. We report best-epoch Top-1 separately as the maximum over the 300 recorded validation epochs. It uses a different checkpoint from final Top-5.

Table 11: ViT-Tiny/ImageNet-1K, 300 epochs. Values are validation accuracies (%). SD denotes sample standard deviation across three seeds. Microbatch times accumulation equals 4096 in every row.

Method	Seed	Microbatch	Accum.	Best Top-1	Final Top-1	Final Top-5
FixedWD	42	512	8	73.666	73.590	91.216
FixedWD	123	512	8	73.842	73.802	91.372
FixedWD	456	512	8	73.806	73.762	91.282
FixedWD	Mean $\pm$ SD	–	–	73.771 ± 0.093	73.718 ± 0.113	91.290 ± 0.078
EqWD	42	1024	4	73.700	73.700	91.310
EqWD	123	512	8	73.898	73.808	91.276
EqWD	456	512	8	73.784	73.710	91.088
EqWD	Mean $\pm$ SD	–	–	73.794 ± 0.099	73.739 ± 0.060	91.225 ± 0.120

The paired EqWD-minus-FixedWD differences for final Top-1 are $+0.110$, $+0.006$, and -0.052 points for seeds 42, 123, and 456, respectively; their mean and sample standard deviation are $+0.021 \pm 0.082$ points. For best-epoch Top-1, the paired differences are $+0.034$, $+0.056$, and -0.022 points, averaging $+0.023 \pm 0.040$. For final Top-5, they are $+0.094$, -0.096, and -0.194 points, averaging -0.065 ± 0.146. The aggregates use unrounded recorded values and round only for presentation. The evidence is insufficient for a consistent classification-improvement claim.

C.4 DINO OBJECT DETECTION EXPERIMENTS

Tables 12 and 13 list the supplied AP and AR summaries after multiplying the supplied fractions by 100. Overall box AP is the reported mAP quantity. AP_{50} and AP_{75} use their respective IoU thresholds; the remaining columns separate object sizes or detection limits. The evaluator uses at most 100 detections for overall AP and 1000 for the other AP and size-specific metrics, while the model configuration caps predictions at 300 per image. Overall AP is separate from the mean of AP_s, AP_m, and AP_l.

Table 12: Updated DINO bounding-box AP results on COCO2017 (0–100 scale, ↑).

Method	Backbone	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
FixedWD	ResNet-50	48.9	66.4	53.2	31.6	52.1	63.1
EqWD	ResNet-50	49.5	66.3	53.0	31.0	52.5	63.4
FixedWD	Swin-T	51.2	68.8	55.7	34.2	53.8	66.2
EqWD	Swin-T	51.3	69.1	55.6	33.8	54.1	66.2

Table 13: Updated DINO average recall on COCO2017 (0–100 scale, ↑), from the same supplied summary as Table 12.

Method	Backbone	AR_{100}	AR_{300}	AR_{1000}	AR_s	AR_m	AR_l
FixedWD	ResNet-50	72.3	72.7	72.7	56.3	76.7	88.2
EqWD	ResNet-50	72.7	73.1	73.1	55.5	77.4	88.8
FixedWD	Swin-T	74.1	74.4	74.4	58.7	78.2	89.9
EqWD	Swin-T	73.9	74.3	74.3	58.6	78.2	89.8

With ResNet-50, overall AP increases by 0.6 points, AP_m by 0.4, and AP_l by 0.3, while AP_{50}, AP_{75}, and AP_s decrease by 0.1, 0.2, and 0.6 points. Overall AR increases by 0.4 points at all three reported detection limits, but AR_s decreases by 0.8 points. With Swin-T, overall AP increases by 0.1 points, AP_{50} by 0.3, and AP_m by 0.3. AP_{75} and AP_s decrease by 0.1 and 0.4, while EqWD and FixedWD have the same AP_l at the reported precision. AR_{100} decreases by 0.2 points, AR_{300} and AR_{1000} by 0.1, AR_s and AR_l by 0.1, while AR_m stays equal. The overall AP gains therefore leave uniform improvements in localization, small-object detection, or recall unsupported.

D DIAGNOSTICS, RUNTIME, AND REPRODUCIBILITY

D.1 DIAGNOSTIC LOGGING AND TRAJECTORY EXTRACTION

The training code logs diagnostics after manual decay and before the base optimizer update. The logger recomputes its field `r_t` from the already-decayed parameters, but copies the mean, residual scale, and signed standardized residual from the controller's pre-decay state. Plotting these fields without accounting for their order would mix different parameter states. For records with controller state, we reconstruct the detector input as

$$r_{t,\mathrm{detector}} = \mu_t + z_t\big(\sigma_{\mathrm{eff},t} + \varepsilon\big),$$

where the source records $z_t = (r_{t,\mathrm{detector}} - \mu_t)/(\sigma_{\mathrm{eff},t} + \varepsilon)$. The extraction keeps both logged and reconstructed ratios and marks nonfinite or unavailable values. At overflow steps, stale state can prevent reconstruction of a fresh observation; the extracted curves cannot certify the absence of overflow.

The original logger samples every 50 steps on CIFAR and 200 on ImageNet. The figure extract retains every tenth logged point for three fixed tensors: `conv1.weight`, `layer2.0.conv1.weight`, and `fc.weight`. The three tensors cover the initial convolution, an intermediate residual block, and the classifier. The plotted traces use seed 42; the archive includes these same tensors for all three seeds. Epoch and phase summaries use every logged eligible-tensor record, not only the plotted points. The modulation maxima refer only to this logged sample; unobserved steps are outside the summary.

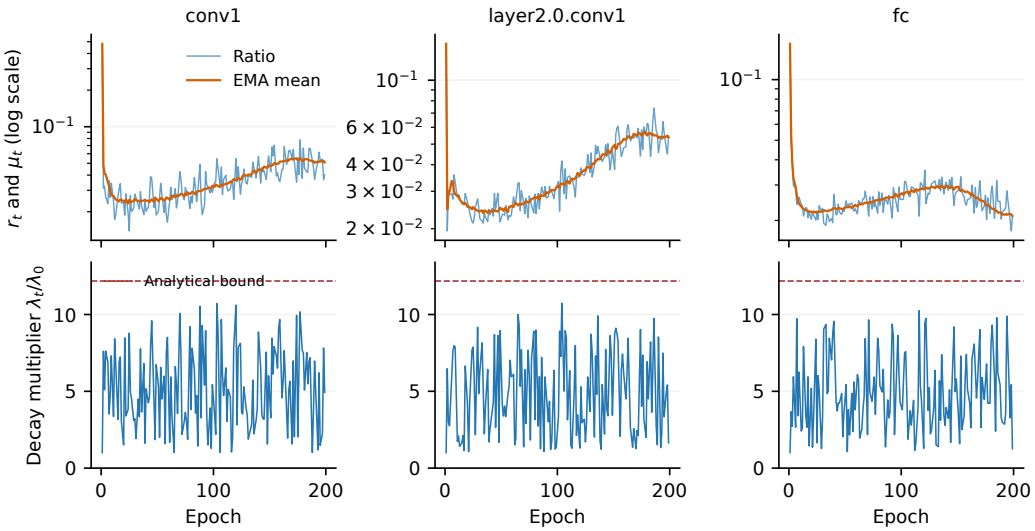

Figure 2: CIFAR-100 EqWD trajectories retain modulation throughout training. Seed 42; the same three tensor names and extraction rule as the ImageNet figure. The horizontal dashed line gives the analytical coefficient bound.

Table 14: Modulation summaries over all logged tensors and three seeds. Epoch intervals are inclusive. P99 is the empirical 99th percentile of the recorded coefficient multipliers.

Benchmark	Epochs	Mean multiplier	P99
CIFAR-100	1–20	4.955	10.401
	21–80	4.938	10.471
	81–160	5.038	10.600
	161–200	4.824	10.299
ImageNet-1K	1–30	1.783	3.573
	31–60	1.778	3.533
	61–80	1.798	3.486
	81–90	1.769	3.631

D.2 Runtime measurement

The controlled benchmark uses Python 3.13.12, PyTorch 2.8.0 with CUDA 12.8, and one NVIDIA L20X. Its fixed synthetic minibatches have sizes 128 for ResNet-20 and 256 for ResNet-50. Both methods use learning rate 0.01, SGD momentum 0.9, and $\lambda_0 = 5 \cdot 10^{-4}$. The benchmark EqWD setting is $(\beta, \alpha_\mu, \alpha_v, \sigma_{\min}) = (1, 0.9, 0.9, 10^{-4})$. The timing experiment therefore measures this implementation at a common setting. It excludes the full input pipeline and each benchmark's selected hyperparameters. The benchmark synchronizes CUDA around timing, interleaves five repeats per method, and measures 500 steps after 100 warm-up steps. It excludes diagnostic logging and data loading. The recorded peak allocated GPU memory is the same between methods within each architecture; ResNet-50 reaches 11,260.23 MiB. This measurement does not show that the extra scalar states have zero memory cost.

Individual training elapsed times remain in the archive for reference. Those runs used concurrent device allocations, so the overhead comparison uses the dedicated benchmark above.

D.3 Archived records and verification

The accompanying archive contains 126 completed training result files with epoch histories, the runtime measurements, budget derivations, selection records, and source snapshots. A manifest records source-relative paths and SHA-256 checksums. The verification script checks archived bytes,

epoch counts, finite accuracy values, equality of each reported best value to the history maximum, and equality of each final value to the last epoch. It recomputes sample standard deviations and runtime medians, reproducing the summaries from saved records.

Trajectory extracts retain hashes of the full diagnostic files. The source snapshots document the implementation used for the completed runs. The archive records the training histories, configurations, and verification inputs used to reproduce the reported summaries.