DeltaMomentum: A Key-Value based Anisotropic Momentum Update via Delta Rule

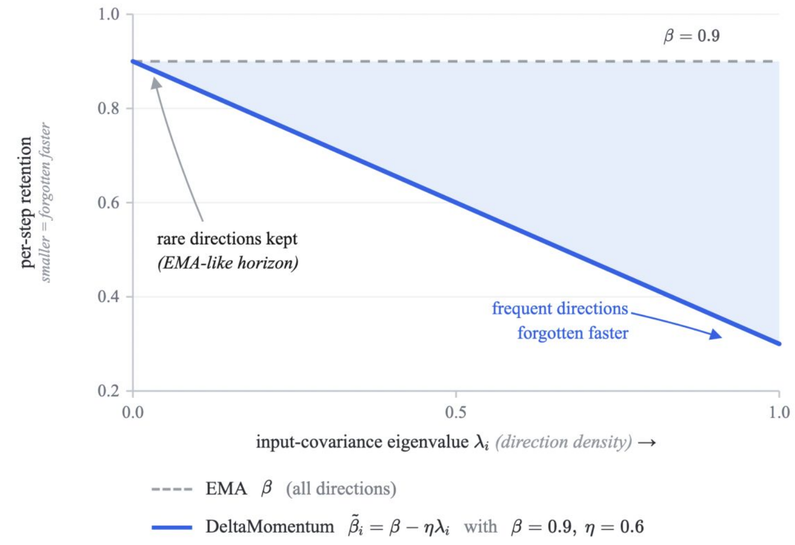
Euijin Hong[1], Guannan Qu[1]

[1]Carnegie Mellon University

Motivation

Standard momentum with exponential moving average (EMA) update rule has single factor β as a “memory horizon” of gradients. However, each eigendirection of the input **requires different forgetting rate**, based on their appearance rate along training.

- High appearance rate directions require faster forgetting
- Low appearance rate directions require slower forgetting



Core Methodology

EMA momentum

$$M_t = \beta M_{t-1} + (1-\beta)\, g_t$$

every direction fades at one rate β

DeltaMomentum *(ours)*

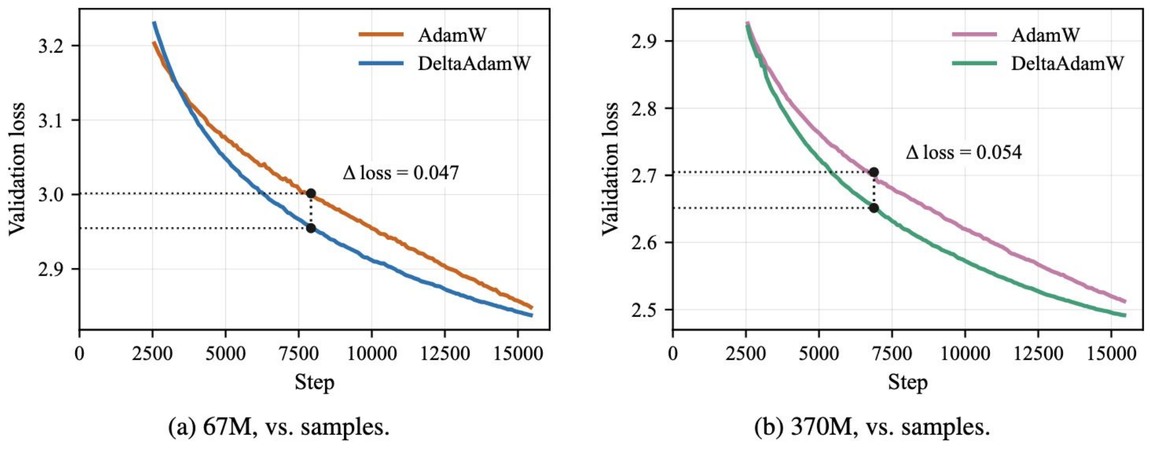
$$M_t = \beta M_{t-1} + \eta \underbrace{\delta_t\, x_t^\top}_{\text{write the new } \delta_t} - \eta \underbrace{M_{t-1} x_t\, x_t^\top}_{\text{erase along } x_t}$$

β is the global forget rate — identical to EMA momentum

Factoring gives M□ = M□₋₁(βI − ηx□x□ᵀ) + ηδ□x□ᵀ, and in expectation **E[βI − ηxxᵀ] = βI − ηΣₓ**, whose eigenvalues are **β̃ᵢ = β − ηλᵢ.**

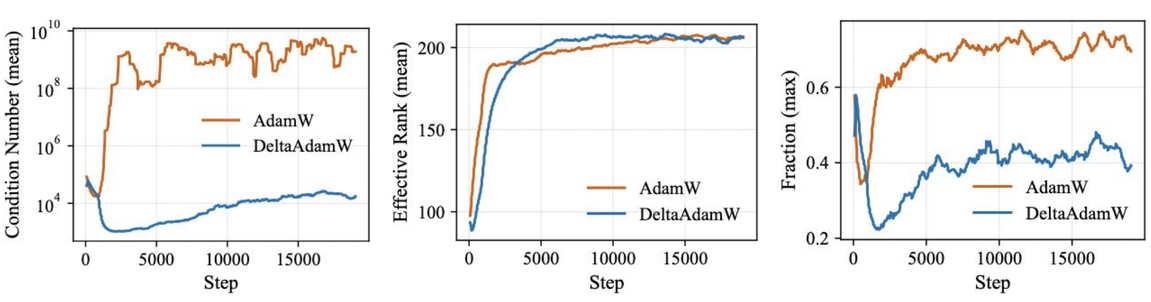
Empirical Results

Language modeling pre-training results



(a) 67M, vs. samples. (b) 370M, vs. samples.

Per-layer input-feature covariance



(a) Condition number (mean). **(b)** Effective rank (mean). **(c)** Top-1 variance fraction (max).

Contribution

- First treatment of the first-moment buffer as an online associative memory of the factored, input-keyed gradient.
- Delta rule-based update enables **key-induced memory horizon for each input direction** of the momentum buffer.
- Theoretical derivations:
  - Validity as a momentum.
  - **Inherent curvatures** reflected in weight & momentum update: Tikhonov–Wiener fixed point & Implicit input-side natural-gradient preconditioning (the K-FAC input factor) with no matrix inversion.
  - **Strictly faster clearance of stale directions** compared to that of EMA momentum.
- Practical insights:
  - **Drop-in replacement** for the first-moment in any optimizer.
  - **μP scaling** allows η to be transferred in zero-shot while the base LR rule inherited.

Theoretical Aspects

1) Anisotropic Memory Transport

$$M_t \;=\; \eta \sum_{\ell=1}^{t} \delta_\ell\, x_\ell^\top \underbrace{\prod_{j=\ell+1}^{t} \left(\beta I_n - \eta\, x_j x_j^\top\right)}_{P_{\ell\to t} \in \mathbb{R}^{n\times n}}, \qquad P_{t\to t} = I_n.$$

$$\mathbb{E}[P_{\ell\to t}]\, u_i \;=\; (\beta - \eta\lambda_i)^{t-\ell}\, u_i \;=\; \tilde{\beta}_i^{\,t-\ell}\, u_i, \qquad \tilde{\beta}_i := \beta - \eta\lambda_i.$$

2) Tikhonov-regularized Wiener fixed point

$$M^\star \;=\; \eta\, \bar{G}\left((1-\beta)I + \eta\Sigma_x\right)^{-1}, \quad M^\star u_i = \phi_i\, \bar{G} u_i, \quad \phi_i = \frac{\eta}{(1-\beta)+\eta\lambda_i} = \frac{1}{\mu+\lambda_i},$$

3) KFAC connection

DeltaMomentum implements input-side natural-gradient preconditioning without inverting Σx, at the cost of a single rank-1 update per step.

4) Direction-selective tracking under distribution shift

$$\textit{EMA:} \quad \mathbb{E}[T_{t,i} \mid T_{t-1,i}] \;=\; \beta\, T_{t-1,i}, \quad \forall i,$$
$$\textit{Delta:} \quad \mathbb{E}[T_{t,i} \mid T_{t-1,i}] \;=\; \tilde{\beta}_i\, T_{t-1,i} \;=\; (\beta - \eta\lambda_i)\, T_{t-1,i}.$$

Key Takeaways

DeltaMomentum assigns each direction its own memory horizon, set by how often the data asks for it.

- **Mechanism.** The delta rule on the first moment gives each input direction its own retention $\tilde{\beta}_i = \beta - \eta\lambda_i$ and memory horizon $\tau_i = 1/((1-\beta) + \eta\lambda_i)$. Frequent directions are overwritten fast, rare ones keep an EMA-like horizon.
- **Theory.** The Tikhonov–Wiener fixed point performs input-side natural-gradient preconditioning with no matrix inversion. Tracking error under drift contracts at $\tilde{\beta}_i < \beta$ for every $\lambda_i > 0$. A single η transfers zero-shot across widths under μP.
- **Results.** Up to 21% (67M) and 21.55% (370M) fewer steps to AdamW's validation loss on FineWeb-Edu, lower terminal loss, per-FLOP Pareto superior after ~2.5k steps. Drop-in first moment for any optimizer, zero persistent memory overhead. Composes with Muon, Shampoo, SOAP, and K-FAC.

Paper Link: