Steepest Descent on Manifolds: 6. Muon + Dual Rotation

Su Jianlin

2026-06-08

We know that when updating matrix parameters using optimizers such as Adam or Muon, the singular values and left-right singular vectors all change accordingly, and they are typically coupled. It is precisely due to this coupling that we cannot easily control the singular values of matrix parameters; thus, when singular values grow abnormally, we cannot effectively prevent it, potentially leading to training failure.

Inspired by *Pion: A Spectrum-Preserving Optimizer via Orthogonal Equivalence Transformation* (hereafter referred to as Pion), this article proposes a variant of Muon that separately updates the left and right singular vectors of a matrix—called "Rotational Muon (MuonR)"—which maintains the singular value distribution of the matrix unchanged, thereby ensuring training stability.

Review of Previous Work

Since matrices composed of left and right singular vectors are necessarily orthogonal matrices, we begin by briefly reviewing Muon under orthogonal constraints. Let the parameter $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ satisfy $\boldsymbol{W}^\top\boldsymbol{W} = \boldsymbol{I}$, and let the update be $\Delta\boldsymbol{W} = -\eta\boldsymbol{\Phi}$. We want the parameter to remain orthogonal after the update. The corresponding steepest descent problem under spectral norm is then

$$\max_{\boldsymbol{\Phi}} \operatorname{tr}(\boldsymbol{G}^\top\boldsymbol{\Phi}) \qquad \text{s.t.} \qquad \|\boldsymbol{\Phi}\|_2 = 1, \quad (\boldsymbol{W} - \eta\boldsymbol{\Phi})^\top(\boldsymbol{W} - \eta\boldsymbol{\Phi}) = \boldsymbol{I} \tag{1}$$

where $\operatorname{tr}(\cdot)$ denotes the trace. The solution is $\boldsymbol{\Phi} = \boldsymbol{W}\boldsymbol{O}$, where $\boldsymbol{O} = \operatorname{msign}([\boldsymbol{W}^\top\boldsymbol{G}]_{\text{skew}})$, and $[\boldsymbol{X}]_{\text{skew}} = (\boldsymbol{X} - \boldsymbol{X}^\top)/2$ is the skew-symmetrization operator. With the retraction operation, the full update rule becomes:

$$\boldsymbol{W} \quad \leftarrow \quad \boldsymbol{W}(\boldsymbol{I} - \eta\boldsymbol{O})\left(\boldsymbol{I} - \boldsymbol{O}^\top\boldsymbol{O} + \frac{\boldsymbol{O}^\top\boldsymbol{O}}{\sqrt{1+\eta^2}}\right) \tag{2}$$

In particular, if $[\boldsymbol{W}^\top\boldsymbol{G}]_{\text{skew}}$ is full rank, this simplifies to

$$\boldsymbol{W} \quad \leftarrow \quad \frac{\boldsymbol{W}(\boldsymbol{I} - \eta\boldsymbol{O})}{\sqrt{1+\eta^2}} \tag{3}$$

The derivation can be found in *Steepest Descent on Manifolds: 2. Muon + Orthogonal*, so it will not be detailed here. Both Eq. (2) and Eq. (3) are fully analytical, adding only a few matrix multiplications to Muon without significantly increasing complexity, making the result completely practical.

Now consider a matrix $\boldsymbol{W} \in \mathbb{R}^{n\times m}(n \geq m)$ that satisfies $\boldsymbol{W}^\top\boldsymbol{W} = \boldsymbol{I}$; such a matrix is said to lie on the Stiefel manifold, a generalization of orthogonal matrices. The above results can theoretically be extended to the Stiefel manifold, but in the non-square case, solving a nonlinear system of equations is required, making practical implementation difficult. Details can be found in *Steepest Descent on Manifolds: 3. Muon + Stiefel*.

Instantaneous Re-parameterization

Next, we turn our attention to an arbitrary parameter matrix $\boldsymbol{W} \in \mathbb{R}^{n\times m}$, with the goal of keeping singular values unchanged during parameter updates, thereby eliminating the possibility of singular value explosion.

To achieve this, we adopt the idea of "instantaneous re-parameterization": before starting the update, we re-parameterize $\boldsymbol{W}$ as $\tilde{\boldsymbol{W}} = \boldsymbol{L}\boldsymbol{W}\boldsymbol{R}$, where $\boldsymbol{L} \in \mathbb{R}^{n\times n}, \boldsymbol{R} \in \mathbb{R}^{m\times m}$, initialized as identity matrices. Thus, initially $\tilde{\boldsymbol{W}} = \boldsymbol{W}$, and with $\boldsymbol{G} = \nabla_{\boldsymbol{W}}\mathcal{L}$, we can write

$$\nabla_{\boldsymbol{L}}\mathcal{L} = \boldsymbol{G}\boldsymbol{W}^\top, \qquad \nabla_{\boldsymbol{R}}\mathcal{L} = \boldsymbol{W}^\top\boldsymbol{G} \tag{4}$$

Next, we freeze $\boldsymbol{W}$ and only update $\boldsymbol{L}$ and $\boldsymbol{R}$, while maintaining the orthogonality of $\boldsymbol{L}$ and $\boldsymbol{R}$ during the update. This ensures that the updated $\tilde{\boldsymbol{W}}$ retains the same singular values as $\boldsymbol{W}$. From the perspective of $\boldsymbol{L}$ and $\boldsymbol{R}$, the problem reduces again to steepest descent on an orthogonal manifold. Since $\boldsymbol{L}, \boldsymbol{R}$ are now square matrices, the steepest descent has a fully analytical solution. Using Eq. (3), the update rules are directly written as:

$$\boldsymbol{L} \quad \leftarrow \quad (\boldsymbol{I} - \eta\boldsymbol{O}_L)\left(\boldsymbol{I} - \boldsymbol{O}_L^\top\boldsymbol{O}_L + \frac{\boldsymbol{O}_L^\top\boldsymbol{O}_L}{\sqrt{1+\eta^2}}\right) \tag{5}$$

$$\boldsymbol{R} \quad \leftarrow \quad (\boldsymbol{I} - \eta\boldsymbol{O}_R)\left(\boldsymbol{I} - \boldsymbol{O}_R^\top\boldsymbol{O}_R + \frac{\boldsymbol{O}_R^\top\boldsymbol{O}_R}{\sqrt{1+\eta^2}}\right) \tag{6}$$

where $\boldsymbol{O}_L = \mathrm{msign}([\boldsymbol{G}\boldsymbol{W}^\top]_{\mathrm{skew}})$, $\boldsymbol{O}_R = \mathrm{msign}([\boldsymbol{W}^\top\boldsymbol{G}]_{\mathrm{skew}})$, and $\mathrm{msign}(\boldsymbol{X})$ denotes the matrix sign function, defined as $\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X})^{-1/2}$ when $\boldsymbol{X}$ is full rank. Multiplying the updated $\boldsymbol{L}, \boldsymbol{R}$ with $\boldsymbol{W}$ yields the complete update rule:

$$\boldsymbol{W} \quad \leftarrow \quad \boldsymbol{L}\boldsymbol{W}\boldsymbol{R} \tag{7}$$

This is the "Rotational Muon (Muon under Rotation, MuonR)", derived from the idea of "instantaneous re-parameterization". In practical scenarios, momentum is typically present; we interpret it as the smoothed gradient, so only need to replace the gradient $\boldsymbol{G}$ with the momentum $\boldsymbol{M}$.

Some Details

Since both $\boldsymbol{L}$ and $\boldsymbol{R}$ require one msign computation per update, even in the ideal case ($n = m$), MuonR has twice the computational cost of Muon. However, for sufficiently large models, this doubling in computation has a negligible impact on end-to-end training time and is usually acceptable. To reduce this overhead, one may consider alternating updates of $\boldsymbol{L}$ and $\boldsymbol{R}$ to amortize the computational cost.

In fact, the main drawback of MuonR is that it keeps all singular values of the matrix unchanged throughout training, which means we must fix the full set of singular values at initialization. This is not easy, because matrices at different positions may require different scales, and forcing them to share a single set of values is likely suboptimal.

A feasible approach is, based on appropriate random initialization, to add an element-wise multiplication vector before or after each matrix to compensate for scale degrees of freedom. For matrices immediately following RMSNorm, the gamma parameters inherent in RMSNorm already serve this function, so this operation can be omitted for such matrices.

Regarding how to choose initial singular values, we can adopt conventional random initialization, or construct them according to a Zipf's law-like form. Furthermore, we may attempt to adjust the singular value entropy to match the optimal entropy computed in *Is Higher Singular Value Entropy Better for Matrix Parameters?*, aiming for improved performance.

Of course, if we can indeed determine the singular values of the matrix in advance—such as when we expect certain parameters to remain orthogonal—then these considerations are unnecessary, and we can directly apply MuonR.

Mid-training Switching

Another optional approach is "mid-training switching", using MuonR only as a "stabilization" mechanism.

Specifically, we start with standard Muon and monitor the spectral norm or Frobenius norm of the matrix. Once the norm exceeds the desired range, we switch to MuonR. Since both variants of Muon rely only on the same gradient/momentum, differing only in computation, such switching is permissible. Because MuonR does not change singular values, neither the spectral nor Frobenius norms will grow further, making it ideal for stabilization.

However, care must be taken to align the update magnitudes before and after switching, to avoid introducing "jumps". To this end, we consider the first-order approximation of MuonR:

$$\boldsymbol{L}\boldsymbol{W}\boldsymbol{R} \approx (\boldsymbol{I} - \eta\boldsymbol{O}_L)\boldsymbol{W}(\boldsymbol{I} - \eta\boldsymbol{O}_R) \approx \boldsymbol{W} - \eta(\boldsymbol{O}_L\boldsymbol{W} + \boldsymbol{W}\boldsymbol{O}_R) \tag{8}$$

Since the singular values of $\boldsymbol{O}_L, \boldsymbol{O}_R$ do not exceed 1 (note that we cannot guarantee $[\boldsymbol{G}\boldsymbol{W}^\top]_{\text{skew}}$ and $[\boldsymbol{W}^\top\boldsymbol{G}]_{\text{skew}}$ are full rank, so we cannot directly use the orthogonality

of $\boldsymbol{O}_L, \boldsymbol{O}_R$), we have $\|\boldsymbol{O}_L\boldsymbol{W}\|_F \leq \|\boldsymbol{W}\|_F$ and $\|\boldsymbol{W}\boldsymbol{O}_R\|_F \leq \|\boldsymbol{W}\|_F$, so:

$$\|\boldsymbol{O}_L\boldsymbol{W} + \boldsymbol{W}\boldsymbol{O}_R\|_F \leq \|\boldsymbol{O}_L\boldsymbol{W}\|_F + \|\boldsymbol{W}\boldsymbol{O}_R\|_F \leq 2\|\boldsymbol{W}\|_F \tag{9}$$

Standard Muon uses $\boldsymbol{W} - \eta\,\mathrm{msign}(\boldsymbol{G})$, and the Frobenius norm of $\mathrm{msign}(\boldsymbol{G})$ is typically $\sqrt{\min(n,m)}$. Therefore, to align the Frobenius norms of the update steps when switching from Muon to MuonR, the learning rate should roughly be multiplied by $\frac{\sqrt{\min(n,m)}}{2\|\boldsymbol{W}\|_F}$.

In practice, the first inequality above may not be tight; $\boldsymbol{O}_L\boldsymbol{W}$ and $\boldsymbol{W}\boldsymbol{O}_R$ are often nearly orthogonal, so by the Pythagorean theorem, the result should be approximately $\sqrt{2}\|\boldsymbol{W}\|_F$, suggesting the multiplier should be increased by a factor of $\sqrt{2}$. However, since $\sqrt{2}$ is not drastically different from 1, and to ensure robustness in extreme cases, it is advisable to retain the original form.

Comparative Analysis

As stated at the beginning, MuonR is inspired by Pion. Let us now examine their connections and differences.

First, the key idea of restricting updates to a dual-rotation form—left and right multiplication by orthogonal matrices—originates from Pion. Once this update form is chosen, obtaining the gradients via "instantaneous re-parameterization" is fairly natural. After this point, Pion and MuonR diverge:

1. Pion uses matrix exponentials of antisymmetric matrices to enforce orthogonality, approximated up to second order in practice.
2. Pion follows the Adam approach, maintaining separate moving averages for the gradients of $\boldsymbol{L}$ and $\boldsymbol{R}$, resulting in four sets of cached variables.
3. MuonR follows the Muon approach, caching only momentum like standard Muon, allowing seamless switching between Muon and MuonR.
4. MuonR is based on the analytical solution of steepest descent on the orthogonal manifold, and achieves orthogonality precisely with only finite additional steps.

Overall, Pion's orthogonality design is relatively heuristic, and the four sets of cache variables may appear daunting. In contrast, MuonR is a relatively natural outcome of works on Muon and steepest descent on orthogonal manifolds, which, in the author's opinion, better adheres to first principles.

Summary

This article proposes MuonR, a variant of Muon that constrains updates to left and right orthogonal rotations. It preserves the singular value distribution of matrices and provides a simple solution for maintaining training stability.