CONTINUOUS APPROXIMATION FOR GRADIENT-BASED OPTIMIZATION: ARBITRARY-ORDER DISCRETIZATION ERROR AND IMPLICIT BIAS

Anonymous authors
Paper under double-blind review

ABSTRACT

Gradient-based optimization methods are the backbone of modern machine learning and deep learning, and thus the theoretical understanding of their dynamics and implicit biases has been crucial recently. A promising line of works in this direction leverages continuous-time approximations to capture the behavior of discrete and iterative updates, shedding light on their convergence, stability, and regularization properties. However, these approximations typically suffer from discretization errors that could become significant with large learning rates. To this end, we propose a unified framework for constructing continuous-time approximations with discretization error to *arbitrary order* of learning rate. This framework can be applied to a wide range of gradient-based methods, including momentum methods and adaptive variants like Adam. This is achieved by segmented Taylor expansions and parameter-splitting techniques that generalize prior works of standard gradient descent, hence enabling a precise analysis of more delicate optimizer dynamics. As a key case study, we derive the first third-order continuous approximation of Adam, which exhibits novel implicit regularization terms that cannot be obtained from lower-order ones. This allows us to identify a new gradual regime transition occurring in Adam's training, explaining its acceleration and generalization behavior, respectively.

1 INTRODUCTION

Gradient-based optimization methods underpin modern machine learning and deep learning by enabling efficient training through iterative computation of loss gradients. Classical algorithms such as stochastic gradient descent (SGD) (Robbins & Monro, 1951), momentum or heavy-ball (HB) (Polyak, 1964), and adaptive methods like Adam (Kingma & Ba, 2017) are indispensable in practice, supporting rapid convergence and robust generalization across diverse tasks. Extensive theoretical work on these methods has analyzed convergence guarantees (Bottou et al., 2018), robustness to noise (Vivien et al., 2022), and adaptive learning-rate mechanisms (Reddi et al., 2018), providing guidance for algorithm design. At the same time, seminal studies on implicit regularization and implicit bias (Soudry et al., 2018; Wilson et al., 2018) show that gradient-based optimizers favor particular low-complexity or margin-maximizing solutions, with direct implications for generalization. Understanding the dynamical behavior and implicit bias of these methods, beyond classical convergence analyses, is therefore crucial for explaining and improving their empirical performance.

Existing research employs several analytical frameworks for gradient-based methods. One line of work directly studies the discrete updates and develops convergence and stability analyses, which are often technically delicate and method-specific (Lessard et al., 2016; Hu & Lessard, 2017; Moalemi & Forbes, 2024). Another line, which we follow, leverages dynamical systems theory to construct *continuous approximations* of the discrete update trajectories. In this setting, the iterates are modeled by explicit ordinary differential equations (ODEs) whose solutions approximate the discrete sequence, enabling the use of calculus-based tools. Continuous approximations have been applied to study convergence and acceleration (Krichene et al., 2015; Krichene & Bartlett, 2017; Xu et al., 2018a;b; Su et al., 2016), stability (Scieur et al., 2017; Ahn et al., 2022), limiting states (Feng & Tu, 2021; Ji & Telgarsky, 2019; Ma et al., 2022), and implicit bias and implicit regularization (Vivien et al., 2022;

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.

Ali et al., 2020; Pesme et al., 2021; Ghosh et al., 2023; Miyagawa, 2023; Cattaneo et al., 2024; Lyu et al., 2025).

A key difficulty in using continuous approximations is the *discretization error* (Eq. (2)) between the continuous trajectory and the discrete iterates, which can be significant for practical learning rates. Recent works employ backward error analysis (BEA) (Hairer et al., 2006; Barrett & Dherin, 2022) to construct modified ODEs whose solutions more closely match the discrete trajectories and thereby reduce this error. For gradient descent (GD), such BEA-based constructions can achieve discretization error of arbitrary order in the learning rate (Miyagawa, 2023; Rosca et al., 2023) and reveal refined implicit regularization effects. For momentum and adaptive variants of GD (e.g., HB, AdaGrad, Adam), however, existing works only obtain continuous approximations with discretization error controlled up to second order (Ghosh et al., 2023; Cattaneo et al., 2024), due to the difficulty of extending BEA to these more complex updates.

To address this limitation, we propose a unified framework that constructs continuous approximations for gradient-based methods with discretization error controlled to arbitrary order in the learning rate. We introduce a segmented continuous approximation and control the discretization error via the number of truncated terms in a Taylor expansion. Our approach is inspired by BEA but differs in that we directly obtain coefficient relations of arbitrary order rather than working with a fixed truncation level. This framework substantially generalizes Miyagawa (2023); Lyu et al. (2025), which focus only on GD and HB. We extend it to a more general Theorem 1 in Section 2 and provide methods for handling optimizers with both momentum and adaptive learning rates (e.g., Adam). This essentially covers mainstream neural-network optimizers and demonstrates the generality of the framework. In particular, our framework yields a concrete third-order continuous approximation of Adam with discretization error $\mathcal{O}(\eta^3)$ beyond the reach of previous analyses. We then use this higher-order continuous approximation to identify additional implicit bias and implicit regularization effects of Adam and provide new insights into its behavior.

Contributions Our main contributions are as follows:

- A general framework is developed for constructing continuous-time approximations of gradient-based methods with discretization error to arbitrary order of the learning rate.
- The framework explicitly accounts for momentum and introduces a parameter-splitting and linearization technique that facilitates extension to adaptive optimizers.
- This framework is applied to derive a third-order continuous-time approximation for Adam, and accurately identifies a novel implicit regularization, i.e. a phase transition phenomenon along the trajectory of Adam explaining its acceleration and generalization behavior, respectively.

Organization. Section 1 presents the overall motivation, related works, and notation. Section 2 introduces our unified framework for constructing continuous approximations with arbitrary-order discretization error and states the main theoretical results, together with the resulting continuous approximations for GD, HB, and Adam. Section 3 applies the framework to Adam, derives a third-order continuous approximation, and analyzes the resulting implicit bias, implicit regularization, and phase-transition behavior, including supporting experiments. Section 4 concludes with a brief discussion.

1.1 Related Works

Continuous Approximations of Gradient-based Optimizers. Continuous approximations are now standard tools for analyzing discrete optimization algorithms (Elkabetz & Cohen, 2021). The simplest constructions assume a small learning rate and apply a first-order Taylor expansion, leading to gradient flow (GF), rescaled GF, and sign-GF for GD, HB, and Adam-type methods, respectively (Kovachki & Stuart, 2021; Ma et al., 2022). These continuous approximations provide useful qualitative insight into the optimizer dynamics but offer only limited, low-order control of the discretization error, which can be inadequate at practical step sizes.

To address this, several works import ideas from BEA (Hairer et al., 2006). For GD, BEA-based constructions yield modified ODEs that match the discrete trajectory up to arbitrarily high order in

the learning rate and reveal refined implicit regularization effects (Barrett & Dherin, 2022; Miyagawa, 2023; Rosca et al., 2023). For HB, Ghosh et al. (2023) derive second-order continuous approximations and Lyu et al. (2025) further give arbitrarily high order approximations. In addition, for Adam, Cattaneo et al. (2024) construct second-order continuous approximations, the best known result currently, and study the associated regularization effect.

Compared to these works, we provide a unified framework that recovers these results as special cases while systematically extending them to arbitrary order and to adaptive gradient methods.

Implicit Bias of Gradient-based Optimizers. Another important topic for optimizers is implicit bias, i.e., the tendency to favor specific solutions without explicit regularization. Continuous approximation is one of the most fundamental tools for analyzing the implicit bias of optimizers in machine learning. There are two main directions in this area: **1)** analyzing the implicit bias of optimization methods for specific datasets and model architectures (Soudry et al., 2018; Azulay et al., 2021; Even et al., 2023; Woodworth et al., 2020; Lyu & Zhu, 2023); and **2)** studying the implicit bias of optimization methods in a model-agnostic way (Jelassi & Li, 2022; Cattaneo et al., 2024).

The implicit bias studied in this paper belongs to the second, model-agnostic category. In this direction, Barrett & Dherin (2022) show that GD prefers solutions with small gradient norm, and Ghosh et al. (2023) reveal an even stronger gradient-norm preference for HB, both derived from implicit regularization terms in their continuous approximations. Dereich et al. (2025) utilized the first-order continuous approximation of Adam to prove that its trajectory converges to the flow of Adam vector field. Using a second-order continuous approximation of Adam, Cattaneo et al. (2024) show that, when the stability constant ε is small, Adam prefers larger or smaller ℓ_1-norm of the local gradient depending on whether $\alpha > \beta$ (β and α are the first and second moment exponential decay rate respectively), whereas for large ε it prefers local gradients with small ℓ_2-norm.

These results motivate us to further investigate Adam's implicit regularization using higher-order continuous approximations. Applying our unified framework, we obtain a general arbitrary-order continuous approximation for Adam and explicitly derive the third-order case. This reveals a new aspect of Adam's implicit bias that is invisible at second order: Adam exhibits two distinct regularization effects with an intriguing phase transition between them during optimization, highlighting the significance of higher-order continuous approximations.

Notations and Conventions. We use $[N]$ for the set of integers $\{0, \dots, N\}$ when N is a positive integer. Given a model $f(\boldsymbol{\theta})$ with parameter $\boldsymbol{\theta} \in \mathbb{R}^d$ and a gradient-based optimizer G, we write $\boldsymbol{\theta}_k$ for the parameter at the k-th iteration of G, and $\boldsymbol{\theta}(t)$ for the solution of a continuous approximation (an ODE). The i-th component of $\boldsymbol{\theta}$ is denoted by θ^i. For vectors with superscript, such as $\boldsymbol{\gamma}^{(1)}$, the i-th component is written as $\gamma^{(1),i}$. Unless otherwise specified, we use $\nabla = \sum_i \mathbf{e}_i \, \partial/\partial\theta^i = \sum_i \mathbf{e}_i \nabla_i$, where $\mathbf{e}_i$ denotes the standard orthonormal basis of $\mathbb{R}^d$. For the loss function $L(\boldsymbol{\theta})$, we let $\mathbf{g}(\boldsymbol{\theta}) := \nabla L(\boldsymbol{\theta})$ and $g^i = \partial L/\partial\theta^i$, and write $\mathbf{g}_k := \nabla L(\boldsymbol{\theta}_k)$ at iteration k. The time derivative of $\boldsymbol{\theta}$ is denoted by $\dot{\boldsymbol{\theta}} = \mathrm{d}\boldsymbol{\theta}/\,\mathrm{d}t$. The inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is $\boldsymbol{a} \cdot \boldsymbol{b}$, and, unless otherwise indicated, vector operations are understood element-wise. We assume throughout that L is infinitely differentiable and that derivatives of all orders are bounded on any closed region. For Adam, we mainly work with the inside-ε variant for convenience, but our conclusions also apply to the outside-ε form, which can be obtained by replacing $\sqrt{\mathbf{g}^2 + \varepsilon}$ with $\sqrt{\mathbf{g}^2} + \varepsilon$ in the equations.

2 A Unified Framework for Constructing Continuous Approximations for Gradient-based Methods

We examine a general formulation that covers a wide range of discrete gradient-based optimizers:

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} - \eta \boldsymbol{U}_k(\nabla L; \boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_k), \tag{1}$$

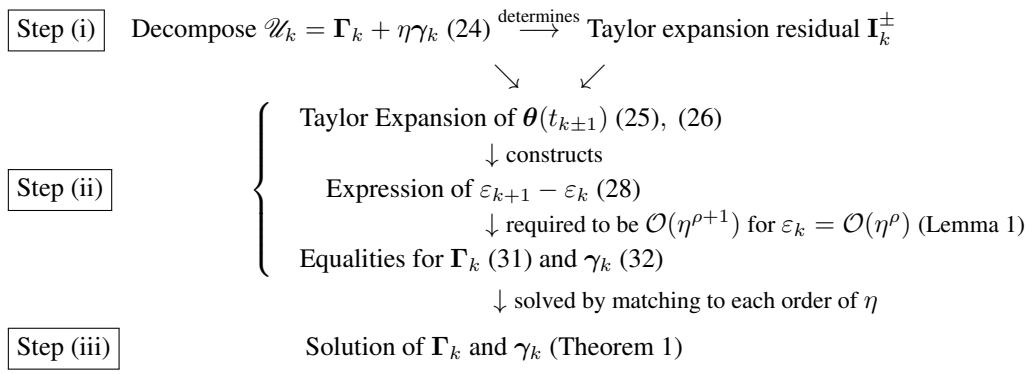
where η is the learning rate and $\boldsymbol{U}_k$ is the update depending on the gradient of the objective function L and the optimization trajectory up to the k-th iteration. For example, for standard gradient descent $\boldsymbol{U}_k$ has the simple form $\boldsymbol{U}_k = \nabla L(\boldsymbol{\theta}_k)$, while for Polyak's Momentum (Heavy Ball Method) $\boldsymbol{U}_k = \nabla L(\boldsymbol{\theta}_k) - \beta(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1})$, where β is the constant momentum factor.

Let $\boldsymbol{\theta}(t) : [0, +\infty) \to \mathbb{R}^d$ be a continuous trajectory obtained from solving the ODE $\dot{\boldsymbol{\theta}}(t) = -\mathscr{U}$, which is expected to be close to the discrete $\boldsymbol{\theta}_k$. Denoting $t_k = k\eta$, we define the *discretization error* of this continuous approximation as

$$\textbf{Discretization Error: } \forall k \in [N] : \ \varepsilon_k := \boldsymbol{\theta}(t_k) - \boldsymbol{\theta}_k. \tag{2}$$

Specifically, $\varepsilon_k = \mathcal{O}(\eta^\rho)$ means that $\boldsymbol{\theta}(t)$ is an order $\mathcal{O}(\eta^\rho)$-close continuous approximation of the discrete optimization Eq. (1). Below we first explain the intuition behind our approach.

Intuition The continuous approximations of Eq. (1) with ε_k controlled to the lowest order of η (e.g., GF for GD) are easy to derive. To obtain higher-order ones, we can start from such a lowest-order continuous approximation and add suitable counter terms to progressively reduce the discretization error, until the desired order of ε_k is achieved. In line with this idea, our approach consists of three key steps: (i) decompose $\mathscr{U}$ into two parts, $\boldsymbol{\Gamma}$ and $\boldsymbol{\gamma}$, so that $\mathscr{U} = \boldsymbol{\Gamma} + \eta\boldsymbol{\gamma}$, where $\dot{\boldsymbol{\theta}} = -\boldsymbol{\Gamma}$ recovers the lowest-order continuous approximation and $\boldsymbol{\gamma}$ collects all counter terms; (ii) derive a set of equalities that $\boldsymbol{\Gamma}$ and $\boldsymbol{\gamma}$ must satisfy when ε_k is required to be of order $\mathcal{O}(\eta^\rho)$ for any $\rho \in \mathbb{Z}^+$; and (iii) solve these equalities to obtain explicit formulas for both $\boldsymbol{\Gamma}$ and $\boldsymbol{\gamma}$, which together define the desired continuous approximation. This approach is explained in detail in Appendix A.1 (summarized in the figure below).



In the following, we apply this intuition to develop a general framework for building continuous approximations for a variety of optimization methods.

In order to obtain an expression for $\boldsymbol{\theta}(t)$ that satisfies our requirements for the discretization error, a straightforward approach can rely on Backward Error Analysis (BEA) method (Hairer et al., 2006). BEA first establishes a series of functions $\boldsymbol{\gamma}^{(\sigma)} : \mathbb{R}^d \to \mathbb{R}^d$ for $\sigma \in \mathbb{N}$ such that

$$\dot{\boldsymbol{\theta}}(t) = -\mathscr{U} = -\sum_{\sigma=0}^{+\infty} \eta^\sigma \boldsymbol{\gamma}^{(\sigma)}. \tag{3}$$

Meanwhile, Taylor expansion can be applied to Eq. (2) at time $t = k\eta$ in the form

$$\boldsymbol{\theta}((k+1)\eta) = \sum_{\sigma=0}^{+\infty} \frac{\eta^\sigma}{\sigma!} \left.\frac{\mathrm{d}^\sigma \boldsymbol{\theta}}{\mathrm{d}t^\sigma}\right|_{t=k\eta}. \tag{4}$$

The core of BEA is now matching the power of η in Eq. (3) and $\boldsymbol{\theta}((k+1)\eta)$ up to a certain ρ-order. Then solving $\boldsymbol{\gamma}^{(\sigma)}$ for $\sigma = 0, 1, 2, \cdots, \rho-1$ can give us the corresponding continuous approximation.

We base our work on this idea, while we consider $\sigma \to \infty$ from the beginning, without being bound to a specific value of ρ. Thus, we will obtain a unified framework for analyzing continuous approximations of arbitrary order for a variety of optimizers. In particular, relying only on Eq. (3) is still insufficient to capture some gradient-based optimization methods. This is because the form of $\boldsymbol{U}_k$ may vary with k, and thus it is important to account for this variability when constructing continuous-time approximations. For instance, in momentum-based optimizers, $\boldsymbol{U}_k$ typically represents a weighted moving average of past gradients, whereas in adaptive methods such as Adam, $\boldsymbol{U}_k$ often incorporates both a rescaled learning rate and historical gradient statistics. These differences imply that $\boldsymbol{U}_k$ evolves in a manner dependent on the optimization algorithm's internal state. Consequently, it

is natural to model the dynamics using *segmented ODEs*, where each segment captures the evolution governed by a specific form of $\boldsymbol{U}_k$.

Specifically, for the optimization process with N total iterations and the corresponding series of intervals as $\forall k \in [N] : T_k = [t_k, t_{k+1})$, a segmented continuous approximation $\boldsymbol{\theta}(t)$ could be expressed as functions defined on these intervals:

$$\dot{\boldsymbol{\theta}}(t) = -\mathscr{U}_k(\nabla L; \boldsymbol{\theta}_0, \cdots, \boldsymbol{\theta}_k), \quad t \in T_k. \tag{5}$$

Then the goal of our framework is to find the explicit formulation of $\mathscr{U}_k$ such that the discretization error is controlled to the desired order $\varepsilon_k = \mathcal{O}(\eta^\rho)$ for arbitrary $\rho \geq 1$. Inspired by BEA, we can express $\mathscr{U}_k$ in Eq. (5) as a form of a series of η

$$\dot{\boldsymbol{\theta}}(t) = -\mathscr{U}_k := -\sum_{\sigma=0}^{+\infty} \eta^\sigma \boldsymbol{\gamma}_k^{(\sigma)}, \quad t \in T_k, \tag{6}$$

where $\gamma_k^{(0)}$ should degenerate to the lowest-order continuous approximation of Eq. (1) and the order-σ term η^σ is the coefficient of the vector $\boldsymbol{\gamma}_k^{(\sigma)}$ representing the update direction on the interval T_k. We now summarize our framework in the following Theorem.

Theorem 1 (General Form of Arbitrary Order Continuous Approximation)**.** *Let k be the iteration number, η be the learning rate, and $t_k = k\eta$. For gradient-based methods that can be written as*

$$\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k = \mu_k(\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}) - \eta \boldsymbol{f}_k(\mathbf{g}_k), \tag{7}$$

the following segmented continuous approximation

$$\dot{\boldsymbol{\theta}} = -\boldsymbol{\Gamma}_k - \eta \boldsymbol{\gamma}_k, \quad t \in [t_k, t_{k+1}),$$

has a discretization error $\mathcal{O}(\eta^\rho)$ for $\rho \geq 1$, if

$$\boldsymbol{\Gamma}_k = \mu_k \boldsymbol{\Gamma}_{k-1} + \boldsymbol{f}_k(\mathbf{g}), \quad \boldsymbol{\gamma}_k = \sum_{\sigma=1}^{\rho-1} \eta^{\sigma-1} \boldsymbol{\gamma}_k^{(\sigma)}, \tag{8}$$

$$\boldsymbol{\gamma}_k^{(\sigma)} = \sum_{j=0}^{k} \left(\prod_{i=j+1}^{k} \mu_i \right) \boldsymbol{\chi}_j^{(\sigma)}, \quad \sigma \in \mathbb{Z}^+, \tag{9}$$

$$\boldsymbol{\chi}_j^{(\sigma)} = \sum_{q=2}^{\sigma+1} \sum_{\mathcal{S}_{\sigma-q+1}^q} \frac{1}{q!} \left[(-1)^q \mathbf{L}_j^{(\sigma_1)} \mathbf{L}_j^{(\sigma_2)} \cdots \mathbf{L}_j^{(\sigma_{q-1})} \boldsymbol{\gamma}_j^{(\sigma_q)} + \mu_j \mathbf{L}_{j-1}^{(\sigma_1)} \mathbf{L}_{j-1}^{(\sigma_2)} \cdots \mathbf{L}_{j-1}^{(\sigma_{q-1})} \boldsymbol{\gamma}_{j-1}^{(\sigma_q)} \right]. \tag{10}$$

Here we define the differential operator $\mathbf{L}_k^{(\sigma)}$ as $\mathbf{L}_k^{(\sigma)} = \boldsymbol{\gamma}_k^{(\sigma)} \cdot \nabla$, and $\mathcal{S}_m^q$ is a set that satisfies the condition

$$\mathcal{S}_m^q = \{(\sigma_1, \sigma_2, \cdots, \sigma_q) | \sum_{p=1}^{q} \sigma_p = m, \sigma_p \in \mathbb{N}\}.$$

Theorem 1 provides us a unified framework to construct first-order ODEs $\dot{\boldsymbol{\theta}} = \boldsymbol{F}_\rho$ that can serve as continuous approximations with discretization error $\mathcal{O}(\eta^\rho)$ for various gradient-based optimization methods, including GD, HB, and Adam. Appendix A provide proofs of the theorem and results.

In Table. 1, we summarize continuous approximations of various optimizers obtained by applying Theorem 1, where we denote $\dot{\theta}^i = \boldsymbol{F}_{\rho,i}$ as the i-th component of $\boldsymbol{F}_\rho \in \mathbb{R}^d$, and define two coefficient scalar functions for convenience $\zeta(x) := \frac{1+10x+x^2}{3(1+x)^2}$, $\psi(x) := \frac{1+x}{1-x}$. Moreover, we define $\omega_1 = \mathbf{g} \cdot \nabla \mathbf{g} \cdot \nabla \mathbf{g}$, $\omega_2 = \mathbf{g} \cdot \nabla(\mathbf{g} \cdot \nabla \mathbf{g})$ for HB and GD, and for Adam

$$\omega_{\beta,1}^i = \frac{\mathbf{g}}{\sqrt{\mathbf{g}^2 + \varepsilon}} \cdot \nabla \mathbf{g} \cdot \frac{\nabla g^i}{\sqrt{\mathbf{g}^2 + \varepsilon}},$$

$$\omega_{\beta,2}^i = \frac{\mathbf{g}}{\sqrt{\mathbf{g}^2 + \varepsilon}} \cdot \nabla \mathbf{g} \cdot \frac{\nabla g^i}{\sqrt{\mathbf{g}^2 + \varepsilon}} + \frac{\mathbf{g}}{\sqrt{\mathbf{g}^2 + \varepsilon}} \cdot \nabla^2 g^i \cdot \frac{\mathbf{g}}{\sqrt{\mathbf{g}^2 + \varepsilon}},$$

$$\omega^i_{\alpha,1} = \sum_u \frac{(g^u)^2 \nabla_u g^i}{((g^u)^2+\varepsilon)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}},$$

$$\omega^i_{\alpha,2} = \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \sum_v \left[\frac{(g^v)^2 \nabla_u g^v \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}} + \frac{3(1-\alpha) g^i g^v \nabla_u g^i \nabla_v g^i}{((g^i)^2+\varepsilon)\sqrt{(g^v)^2+\varepsilon}} - \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{(g^v)^2+\varepsilon}} \right].$$

Table 1: Continuous Approximations of GD, HB and Adam for order $\rho = 1$, 2, and 3.

$\varepsilon_k = \mathcal{O}(\eta^\rho)$	GD	HB	Adam
$\rho = 1$	$F_1^{GD} = -\mathbf{g}$	$F_1^{HB} = -\frac{\mathbf{g}}{1-\beta}$	$F_{1,i}^{AD} = -\frac{g^i}{\sqrt{(g^i)^2+\varepsilon}}$
$\rho = 2$	$F_2^{GD} = F_1^{GD}$ $-\eta \frac{\mathbf{g}\cdot\nabla\mathbf{g}}{2}$	$F_2^{HB} = F_1^{HB}$ $-\eta \frac{1+\beta}{(1-\beta)^3} \frac{\mathbf{g}\cdot\nabla\mathbf{g}}{2}$	$F_{2,i}^{AD} = F_{1,i}^{AD} - \frac{\eta}{2\sqrt{(g^i)^2+\varepsilon}}$ $\times \left[\psi(\beta) - \psi(\alpha)\frac{(g^i)^2}{(g^i)^2+\varepsilon}\right] \frac{\mathbf{g}}{\sqrt{\mathbf{g}^2+\varepsilon}} \cdot \nabla g^i$ e
$\rho = 3$	$F_3^{GD} = F_2^{GD}$ $-\frac{\eta^2}{4}\left[\omega_1 + \frac{\omega_2}{3}\right]$	$F_3^{HB} = F_2^{HB} - \eta^2 \times$ $\frac{(1+\beta)^2}{4(1-\beta)^5}\left[\omega_1 + \zeta(\beta)\omega_2\right]$	$F_{3,i}^{AD} = F_{2,i}^{AD} - \frac{\eta^2}{4} \times$ $\left\{ \frac{(\psi(\beta))^2}{\sqrt{(g^i)^2+\varepsilon}} \left[\omega^i_{\beta,1} + \zeta(\beta)\omega^i_{\beta,2}\right] \right.$ $\left. + \frac{(\psi(\alpha))^2 (g^i)^2}{((g^i)^2+\varepsilon)^{3/2}} \left[\omega^i_{\alpha,1} + \zeta(\alpha)\omega^i_{\alpha,2}\right] \right\}$
Discrete	$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \mathbf{g}_k$	$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \mathbf{g}_k$ $+\beta(\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1})$	$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \frac{\hat{\mathbf{m}}_{k+1}}{\sqrt{\hat{\mathbf{v}}_{k+1}+\varepsilon}}$

3 Continuous Approximation of Adam and Implicit Bias

The iterative update of Adam is given by

$$\textbf{Discrete Adam} \begin{cases} \mathbf{m}_{k+1} = \beta \mathbf{m}_k + (1-\beta)\mathbf{g}_k, & \hat{\mathbf{m}}_{k+1} = \dfrac{\mathbf{m}_{k+1}}{1-\beta^{k+1}} \\ \mathbf{v}_{k+1} = \alpha \mathbf{v}_k + (1-\alpha)\mathbf{g}_k^2, & \hat{\mathbf{v}}_{k+1} = \dfrac{\mathbf{v}_{k+1}}{1-\alpha^{k+1}} \\ \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \dfrac{\hat{\mathbf{m}}_{k+1}}{\sqrt{\hat{\mathbf{v}}_{k+1}+\varepsilon}} \end{cases}, \tag{11}$$

where $\hat{\mathbf{m}}_{k+1}$ and $\hat{\mathbf{v}}_{k+1}$ are unbiased corrections for $\mathbf{m}_{k+1}$ and $\mathbf{v}_{k+1}$ respectively when k is small. Note here that we slightly abuse the notation ε to denote the stability constant in Adam, while we still use ε_k with a subscript k for the discretization error at the k-th iteration.

As k increases, β^{k+1} and α^{k+1} will converge to 0 at a very fast rate. In general, we are more interested in properties such as convergence and implicit bias of an optimizer, so the discretization error of the continuous approximation is more important near its convergent points than the initial iterations. Therefore, we can simplify the computational process by only focusing on the case where k is sufficiently large, ignoring the difference between the variables with and without the hat symbol " $\hat{}$ ". In other words, we will consider the simplified form of Adam

$$\textbf{Discrete Adam with large } k \begin{cases} \mathbf{m}_{k+1} = \beta \mathbf{m}_k + (1-\beta)\mathbf{g}_k \\ \mathbf{v}_{k+1} = \alpha \mathbf{v}_k + (1-\alpha)\mathbf{g}_k^2 \\ \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \dfrac{\mathbf{m}_{k+1}}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}} \end{cases}. \tag{12}$$

Applying our framework to Adam is more complicated than the cases for GD and HB: it is difficult to directly modify the iterative updates shown in Eq. (12) into the linear recursive form Eq. (29), which is required by our framework. The difficulty lies in that the parameters in adjacent updates of Adam

are changing simultaneously: in

$$\Delta_{k+1} = \boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k = -\eta \frac{\mathbf{m}_{k+1}}{\sqrt{\mathbf{v}_{k+1} + \varepsilon}}, \quad \Delta_k = \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1} = -\eta \frac{\mathbf{m}_k}{\sqrt{\mathbf{v}_k + \varepsilon}}, \tag{13}$$

$\mathbf{m}_{k+1}$ changes along with $\mathbf{m}_k$ while $\mathbf{v}_{k+1}$ changes along with $\mathbf{v}_k$. This issue can be addressed as follows. Noting that

$$\begin{aligned} \Delta_{k+1} - \Delta_k =& -\eta \left[\frac{\mathbf{m}_{k+1}}{\sqrt{\mathbf{v}_{k+1} + \varepsilon}} - \frac{\mathbf{m}_k}{\sqrt{\mathbf{v}_{k+1} + \varepsilon}} \right] - \eta \left[\frac{\mathbf{m}_k}{\sqrt{\mathbf{v}_{k+1} + \varepsilon}} - \frac{\mathbf{m}_k}{\sqrt{\mathbf{v}_k + \varepsilon}} \right] \\ =& \left(\Delta_{1,k+1} - \Delta_{1,k} \right) + \left(\Delta_{2,k+1} - \Delta_{2,k} \right). \end{aligned} \tag{14}$$

This means that we can divide Δ_k into two parts: the first part $\Delta_{1,k}$, in which the denominator is a constant (maintained as $\sqrt{\mathbf{v}_{k+1} + \varepsilon}$) and only the numerator is a variable (changed from $\mathbf{m}_k$ to $\mathbf{m}_{k+1}$); the second part $\Delta_{2,k}$, where the numerator is a constant (maintained as $\mathbf{m}_k$) and only the denominator is a variable (changed from $\sqrt{\mathbf{v}_k + \varepsilon}$ to $\sqrt{\mathbf{v}_{k+1} + \varepsilon}$).

We could construct a continuous approximation for the dynamics of $\Delta_{1,k}$ by applying our unified framework proposed in Theorem 1

$$\gamma_{1,k}^{(1),i} \approx \frac{\psi(\beta)}{2\sqrt{\left(g^i\right)^2 + \varepsilon}} \sum_u \frac{g^u \nabla_u g^i}{\sqrt{\left(g^u\right)^2 + \varepsilon}}, \tag{15}$$

$$\gamma_{1,k}^{(2),i} \approx \frac{(\psi(\beta))^2}{4\sqrt{\left(g^i\right)^2 + \varepsilon}} \sum_u \sum_v \left[\frac{\nabla_u g^i \cdot g^v \nabla_v g^u}{\sqrt{\left(g^u\right)^2 + \varepsilon}\sqrt{\left(g^v\right)^2 + \varepsilon}} + \zeta(\beta) \frac{g^u (\nabla_u g^v \nabla_v g^i + g^v \nabla_u \nabla_v g^i)}{\sqrt{\left(g^u\right)^2 + \varepsilon}\sqrt{\left(g^v\right)^2 + \varepsilon}} \right], \tag{16}$$

$$\vdots$$

where $\gamma_{1,k}^{(j),i}$ denotes the i-th component of $\boldsymbol{\gamma}_{1,k}^{(j)}$. A similar process could be also performed for $\Delta_{2,k}$ and obtain

$$\gamma_{2,k}^{(1),i} \approx -\frac{\psi(\alpha)\left(g^i\right)^2}{2\left(\left(g^i\right)^2 + \varepsilon\right)^{3/2}} \sum_u \frac{g^u \nabla_u g^i}{\sqrt{\left(g^u\right)^2 + \varepsilon}} \tag{17}$$

$$\begin{aligned} \gamma_{2,k}^{(2),i} \approx & \frac{(\psi(\alpha))^2 \left(g^i\right)^2}{4\left(\left(g^i\right)^2 + \varepsilon\right)^{3/2}} \left\{ \sum_u \frac{\left(g^u\right)^2 \nabla_u g^i}{\left(\left(g^u\right)^2 + \varepsilon\right)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{\left(g^v\right)^2 + \varepsilon}} \right. \\ & \left. + \zeta(\alpha) \sum_u \frac{g^u}{\sqrt{\left(g^u\right)^2 + \varepsilon}} \sum_v \left[\frac{\left(g^v\right)^2 \nabla_u g^v \nabla_v g^i}{\left(\left(g^v\right)^2 + \varepsilon\right)^{3/2}} + \frac{3(1-\alpha) g^i \nabla_u g^i \cdot g^v \nabla_v g^i}{\left(\left(g^i\right)^2 + \varepsilon\right)\sqrt{\left(g^v\right)^2 + \varepsilon}} - \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{\left(g^v\right)^2 + \varepsilon}} \right] \right\} \end{aligned} \tag{18}$$

$$\vdots$$

Eventually, we summarize the relevant results in the following corollary.

Corollary 1.1 (Arbitrary Order Continuous Approximation of Adam)**.** *For the Adam in Eq.* (12)*, following the framework of Theorem 1, its arbitrary order continuous approximation can be obtained in turn (for large k)*

$$\boldsymbol{\gamma}_k^{(1)} = \boldsymbol{\gamma}_{1,k}^{(1)} + \boldsymbol{\gamma}_{2,k}^{(1)}, \quad \boldsymbol{\gamma}_k^{(2)} = \boldsymbol{\gamma}_{1,k}^{(2)} + \boldsymbol{\gamma}_{2,k}^{(2)}, \quad \cdots$$

in which $\boldsymbol{\gamma}_{1,k}^{(\sigma)}$ (as defined in Eq. (15) *and* (16)*) and $\boldsymbol{\gamma}_{2,k}^{(\sigma)}$ (as defined in Eq.* (17) *and* (18)*) are derived sequentially based on $\boldsymbol{\Gamma}_{1,k}$ and $\boldsymbol{\Gamma}_{2,k}$ respectively, according to Theorem 1.*

Since the higher-order continuous approximation of Adam obtained in this paper does not yet have any other relevant results to serve as a comparison, we will verify the corollary by a numerical experiment on a toy model. Additional regularization based on the third-order continuous approximation will also be verified through experiments, which will in turn confirm the conclusions we have drawn again.

3.1 Discretization Errors of Continuous Approximations of Adam

We empirically verify the continuous approximation of Adam obtained from this framework on a simple 2-d toy model. Firstly, we give the discretization error curves of the continuous approximations of Adam with discretization error to $\mathcal{O}(\eta)$, $\mathcal{O}(\eta^2)$, and $\mathcal{O}(\eta^3)$ under different combinations of hyperparameters. We demonstrate that the convergence points of the continuous approximations are getting closer and closer to those of the original Adam optimizer as the order increases. Secondly, we utilize the results of the continuous approximation to show the effect of the hyperparameters on the location of the convergence point of Adam. The experimental results show that comparing with the $\mathcal{O}(\eta)$ and $\mathcal{O}(\eta^2)$ continuous approximations, the discretization error of our $\mathcal{O}(\eta^3)$ continuous approximation is reduced for all learning rates, both in the case of $\alpha > \beta$ and $\alpha < \beta$. This suggests that our framework is indeed effective in providing higher-order continuous approximations of Adam. Detailed experimental results can be found in Appendix C.1.

3.2 Implicit Bias of Adam

Ma et al. (2022) states that Adam with hyperparameter $\alpha < \beta$ is unstable, so in most practices there is always $\alpha \geq \beta$. This is also the situation that we mainly consider. In Appendix B.1, we will show that this phenomenon may be related precisely to the regularization inspired by the third-order continuous approximation. This is because R_a (defined in Corollary 1.2) in fact consists of a combination of factors containing terms whose sign is determined by the comparison of α and β.

In this work, the third-order continuous approximation of Adam could provide us a novel perspective regarding its implicit bias that cannot be observed in previous works. Particularly, we will reveal two completely different regularization terms induced by the highly accurate approximation, and interestingly there exhibits *a gradual regime transition between these two terms during the optimization*, as shown in the following corollary. The detailed derivation is provided in Appendix B.

Corollary 1.2 (Implicit Bias of Adam from Third-Order Continuous Approximation)**.** *When $\alpha > \beta$, for the third-order continuous approximation of Adam, the additional implicit bias it induces is:*

- *When $\left(g^i\right)^2 \gg \varepsilon$ hold for all g^i, Adam tends to minimize the following term as an implicit bias,*

$$R_a = -\mathrm{sgn}\left(\mathbf{g}\right) \cdot \nabla \mathbf{g} \cdot \mathrm{sgn}\left(\mathbf{g}\right) \tag{19}$$

- *When $\left(g^i\right)^2 \ll \varepsilon$ hold for all g^i, Adam method will minimize the following regularization term implicitly,*

$$R_g = \mathbf{g} \cdot \nabla \mathbf{g} \cdot \mathbf{g}. \tag{20}$$

We summarize each regularization of Adam in Table. 2.

Regime Transition of Adam The regularization terms suggest that when most of the components of gradient is not small enough, the regularization R_a will impel Adam to a region with larger Hessian norm since $\mathbf{H} = \nabla \mathbf{g} = \nabla^2 L$. This indicates that it prefers a sharp region rather than a flat one, dramatically different from the behavior of GD and HB. This preference for sharp regions may be one of the important reasons why Adam converges much faster than GD and HB, so we use "a" for "acceleration" in R_a. However, when most components g^i satisfy $g^i \ll \varepsilon$, the Adam optimizer would shift to another type of regularization R_g, which is the same as HB and encourages the parameters to a flatter region with lower Directional Smoothness (Lyu et al., 2025). Since flat regions typically imply better generalization ability, we use "g" for "generalization" in R_g. We prove by mathematical approximation that Adam degenerates into HB in Appendix A.4.3. Therefore, there should be a significant regime transition in Adam's optimization process. It should be noted that the transition from $g^i \gg \varepsilon$ to $g^i \ll \varepsilon$ does not occur instantaneously; consequently, it results in a gradual regime transition. Experiment 2 in Appendix C.2 illustrates the changes of regularization during the regime transition.

In order to verify the newly discovered regularization effects and regime transition phenomenon, we train LeNet (LeCun et al., 1998) on MNIST with GD, HB and Adam. For detailed information on the experiment, see Appendix C.2. The results confirm a clear regime transition in Adam's third-order regularization that is absent for GD and HB: in the early stage, Adam strongly minimizes R_a while

Table 2: Regularizations of Adam with different orders of approximation. When increasing the order, it will reveal extra regularization terms.

$\varepsilon_k = \mathcal{O}(\eta^\rho)$	continuous approximation	regularization (small ε)	regularization (large ε)
$\rho = 1$	$F_{1,i}^{AD} = -\frac{g^i}{\sqrt{(g^i)^2+\varepsilon}}$	-	-
$\rho = 2$	$F_{2,i}^{AD} = F_{1,i}^{AD} - \frac{\eta}{2\sqrt{(g^i)^2+\varepsilon}} \times \left[\psi(\beta) - \psi(\alpha)\frac{(g^i)^2}{(g^i)^2+\varepsilon}\right] \times \frac{\mathbf{g}}{\sqrt{\mathbf{g}^2+\varepsilon}} \cdot \nabla g^i$	$-\lVert\mathbf{g}\rVert_1$-penalized	$\lVert\mathbf{g}\rVert_2^2$-penalized
$\rho = 3$	$F_{3,i}^{AD} = F_{2,i}^{AD} - \eta^2 \times \left\{\frac{(\psi(\beta))^2}{4\sqrt{(g^i)^2+\varepsilon}} \times \left[\omega_{\beta,1}^i + \zeta(\beta)\omega_{\beta,2}^i\right] + \frac{(\psi(\alpha))^2(g^i)^2}{4\left((g^i)^2+\varepsilon\right)^{3/2}} \times \left[\omega_{\alpha,1}^i + \zeta(\alpha)\omega_{\alpha,2}^i\right]\right\}$	$-\lVert\mathbf{g}\rVert_1$-**penalized and** R_a-**penalized**	$\lVert\mathbf{g}\rVert_2^2$-**penalized and** R_g-**penalized**

R_g remains almost unchanged, whereas around the transition point R_a rapidly loses control and R_g together with the Directional Smoothness decrease, similar to HB. At the same time, the sharp drop in the percentage of coordinates with $\left(g^i\right)^2 > \varepsilon$ aligns closely with the onset of the transition. The experiment therefore provides empirical evidence for the predicted regime transition in Adam's third-order regularization and for the fact that, after the transition, the implicit regularization of Adam becomes essentially the same as that of HB.

Due to the simplicity of the MNIST task, the above experiment alone does not reveal how Adam's regularization affects generalization. To further examine this aspect, following Xiong et al. (2024), we add 10% random label noise to MNIST and repeat the comparison of GD, HB and Adam; the results are summarized in Appendix C.2. In all three configurations of ε, Adam still exhibits the predicted regime transition, and the timing of this transition closely matches the change in its generalization behavior. These observations provide additional evidence that Adam's implicit regularization switches from an R_a-dominated regime to an R_g-dominated regime near minima, and that this switch is closely linked to improved generalization.

4 Discussion

Since Theorem 1 holds independently of ρ, it is theoretically possible to obtain continuous approximations with discretization error to arbitrary order of the learning rate for various optimizers by simply following this framework. The main difficulty of this framework lies in obtaining the linearization relation of the discrete errors of adjacent iterations. This is relatively easy for non-adaptive learning rate methods (for instance, HB), but requires some skill for adaptive learning rate methods like Adam. In this paper, by splitting the difference between the adjacent updates of Adam into two parts with only the numerator changing and only the denominator changing, we obtain a proper linearized approximation and successfully give an expression for the continuous approximation of Adam with discretization error to $\mathcal{O}(\eta^3)$ through our framework, which proves the validity of the framework.

In the future, we will adapt our methodology to analyze the recent optimizers to deepen our understanding of them, e.g. Lion (Chen et al., 2023) and Muon (Liu et al., 2025). However, it should be noted that although numerical experiments have proved the effectiveness of our approach, some of the treatments used in this work are still approximate, and more rigorous ways of treatment are still to be explored. In addition, since the regularization of all orders is always working at the same time in the optimization process, it is difficult to investigate a specific one individually, and a more effective analytical framework is still needed to support this kind of study.

AI USE STATEMENT

In this work, we used generative AI tools to aid writing and polishing. We have not used generative AI tools for other tasks with required disclosure. We have reviewed all AI-assisted work. We take responsibility for the final content of this work.

REFERENCES

Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent, 2022. URL https://arxiv.org/abs/2204.01050.

Alnur Ali, Edgar Dobriban, and Ryan Tibshirani. The implicit regularization of stochastic gradient flow for least squares. In *International Conference on Machine Learning*, volume PMLR, 2020.

Shahar Azulay, Edward Moroshko, Mor Shpigel Nacson, Blake E Woodworth, Nathan Srebro, Amir Globerson, and Daniel Soudry. On the implicit bias of initialization shape: Beyond infinitesimal mirror descent. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 468–477. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/azulay21a.html.

David G. T. Barrett and Benoit Dherin. Implicit gradient regularization. In *International Conference on Machine Learning*, 2022.

Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning, 2018. URL https://arxiv.org/abs/1606.04838.

Matias D Cattaneo, Jason M Klusowski, and Boris Shigida. On the implicit bias of adam. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 5862–5906, 2024.

Xiangning Chen, Chen Liang, Da Huang, Esteban Real, Kaiyuan Wang, Hieu Pham, Xuanyi Dong, Thang Luong, Cho-Jui Hsieh, Yifeng Lu, et al. Symbolic discovery of optimization algorithms. *Advances in neural information processing systems*, 36:49205–49233, 2023.

Steffen Dereich, Arnulf Jentzen, and Sebastian Kassing. Ode approximation for the adam algorithm: General and overparametrized setting. *arXiv preprint arXiv:2511.04622*, 2025.

Omer Elkabetz and Nadav Cohen. Continuous vs. discrete optimization of deep neural networks. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=iX0TSH45eOd.

Mathieu Even, Scott Pesme, Suriya Gunasekar, and Nicolas Flammarion. (s)gd over diagonal linear networks: Implicit regularisation, large stepsizes and edge of stability, 2023.

Yu Feng and Yuhai Tu. The inverse variance–flatness relation in stochastic gradient descent is critical for finding flat minima. *Proceedings of the National Academy of Sciences*, 118(9): e2015617118, 2021. doi: 10.1073/pnas.2015617118. URL https://www.pnas.org/doi/abs/10.1073/pnas.2015617118.

Avrajit Ghosh, He Lyu, Xitong Zhang, and Rongrong Wang. Implicit regularization in heavy-ball momentum accelerated stochastic gradient descent, 2023.

Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric numerical integration*, volume 31. Springer-Verlag, Berlin, second edition, 2006. ISBN 3-540-30663-3; 978-3-540-30663-4.

Bin Hu and Laurent Lessard. Dissipativity theory for nesterov's accelerated method. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 1549–1557. JMLR.org, 2017.

Samy Jelassi and Yuanzhi Li. Towards understanding how momentum improves generalization in deep learning, 2022.

Ziwei Ji and Matus Telgarsky. Gradient descent aligns the layers of deep linear networks. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *arXiv:1412.6980*, 2017.

Nikola B Kovachki and Andrew M Stuart. Continuous time analysis of momentum methods. *JMLR*, 2021.

Walid Krichene and Peter L Bartlett. Acceleration and averaging in stochastic descent dynamics. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/643de7cf7ba769c7466ccbc4adfd7fac-Paper.pdf.

Walid Krichene, Alexandre Bayen, and Peter L Bartlett. Accelerated mirror descent in continuous and discrete time. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/f60bb6bb4c96d4df93c51bd69dcc15a0-Paper.pdf.

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

Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1):57–95, 2016. doi: 10.1137/15M1009597. URL https://doi.org/10.1137/15M1009597.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, et al. Muon is scalable for llm training. *arXiv preprint arXiv:2502.16982*, 2025.

Bochen Lyu and Zhanxing Zhu. Implicit bias of (stochastic) gradient descent for rank-1 linear neural network. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=PjBEUTVzoe.

Bochen Lyu, Xiaojing Zhang, Fangyi Zheng, He Wang, Zheng Wang, and Zhanxing Zhu. Heavy-ball momentum method in continuous time and discretization error analysis, 2025. URL https://arxiv.org/abs/2506.14806.

Chao Ma, Lei Wu, and Weinan E. A qualitative study of the dynamic behavior for adaptive gradient algorithms. In Joan Bruna, Jan Hesthaven, and Lenka Zdeborova (eds.), *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*, volume 145 of *Proceedings of Machine Learning Research*, pp. 671–692. PMLR, 16–19 Aug 2022. URL https://proceedings.mlr.press/v145/ma22a.html.

Taiki Miyagawa. Toward equation of motion for deep neural networks: Continuous-time gradient descent and discretization error analysis. In *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, 2023.

Sepehr Moalemi and James Richard Forbes. Input-output stability of gradient descent: A discrete-time passivity-based approach, 2024. URL https://arxiv.org/abs/2411.16074.

Scott Pesme, Loucas Pillaud-Vivien, and Nicolas Flammarion. Implicit bias of SGD for diagonal linear networks: a provable benefit of stochasticity. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=vvi7KqHQiA.

Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 1964.

Sashank J. Reddi, Satyen Kale, and Surinder Kumar. On the convergence of adam and beyond. *ArXiv*, abs/1904.09237, 2018. URL https://api.semanticscholar.org/CorpusID:3455897.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 1951.

Mihaela Rosca, Yan Wu, Chongli Qin, and Benoit Dherin. On a continuous time model of gradient descent dynamics and instability in deep learning. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=EYrRzKPinA.

Damien Scieur, Vincent Roulet, Francis Bach, and Alexandre d'Aspremont. Integration methods and optimization algorithms. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/bf62768ca46b6c3b5bea9515d1a1fc45-Paper.pdf.

Daniel Soudry, Elad Hoffer, and Mor Shpigel Nacson. The implicit bias of gradient descent on separable data. *JMLR*, 2018.

Weijie Su, Stephen Boyd, and Emmanuel J. Candès. A differential equation for modeling nesterov's accelerated gradient method: Theory and insights. *Journal of Machine Learning Research*, 17 (153):1–43, 2016. URL http://jmlr.org/papers/v17/15-084.html.

Loucas Pillaud Vivien, Julien Reygner, and Nicolas Flammarion. Label noise (stochastic) gradient descent implicitly solves the lasso for quadratic parametrisation. In Po-Ling Loh and Maxim Raginsky (eds.), *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pp. 2127–2159. PMLR, 02–05 Jul 2022. URL https://proceedings.mlr.press/v178/vivien22a.html.

Ashia C. Wilson, Benjamin Recht, and Michael I. Jordan. A lyapunov analysis of momentum methods in optimization, 2018. URL https://arxiv.org/abs/1611.02635.

Blake Woodworth, Suriya Gunasekar, Jason D Lee, Edward Moroshko, Pedro Savarese, Itay Golan, Daniel Soudry, and Nathan Srebro. Kernel and rich regimes in overparametrized models. In *Conference on Learning Theory*, 2020.

Haoyi Xiong, Xuhong Li, Boyang Yu, Dongrui Wu, Zhanxing Zhu, and Dejing Dou. Stochastic gradient descent with random label noises: doubly stochastic models and inference stabilizer. *Machine Learning: Science and Technology*, 5(1):015039, mar 2024. doi: 10.1088/2632-2153/ad13ba. URL https://dx.doi.org/10.1088/2632-2153/ad13ba.

Pan Xu, Tianhao Wang, and Quanquan Gu. Accelerated stochastic mirror descent: From continuous-time dynamics to discrete-time algorithms. In *International Conference on Artificial Intelligence and Statistics*, pp. 1087–1096. PMLR, 2018a.

Pan Xu, Tianhao Wang, and Quanquan Gu. Continuous and discrete-time accelerated stochastic mirror descent for strongly convex functions. In *International Conference on Machine Learning*, pp. 5492–5501. PMLR, 2018b.

APPENDIX

A Proof of Theorems and Corollaries **13**

A.1 Preliminaries 13

A.2 Proof of Theorem 1 15

A.3 Continuous Approximations for GD and HB 18

A.3.1 $\rho = 2$. 20

A.3.2 $\rho = 3$. 20

A.4 Proof of Corollary 1.1 21

A.4.1 $\rho = 2$. 22

A.4.2 $\rho = 3$. 23

A.4.3 Relationship between Adam and GD/HB 28

B Implicit Bias of Gradient-based Methods embodied in higher-order continuous approximations **29**

B.1 Implicit bias of Adam if $\left(g^i\right)^2 \gg \varepsilon$ 29

B.2 Implicit bias of Adam if $\left(\mathbf{g}^i\right)^2 \ll \varepsilon$ 31

C Details for Numerical Experiments **32**

C.1 Details for Section 3.1 32

C.2 Details for Section 3.2 33

A PROOF OF THEOREMS AND COROLLARIES

In Appendix A, we provide proofs for theorems and corollaries in Section 2.

A.1 PRELIMINARIES

Before diving into the proof, in this section, we first give a detailed explanation for the key steps of our approach summarized in the main paper.

Decompose $\mathscr{U}_k$ To make Eq. (6) an $\mathcal{O}(\eta^\rho)$-close continuous approximation of Eq. (1), deriving formulations for each term of $\boldsymbol{\gamma}_k^{(\sigma)}$ is required. While this might seem to be laborious, our procedure (as shown in Theorem 1) fortunately provides us a convenient way to calculate $\boldsymbol{\gamma}_k^{(\sigma)}$ in a recursive manner by employing the relations among them. Hence it is possible to directly derive any higher order continuous approximations from the lower order ones. And the remaining step is to derive the lowest order term $\gamma_k^{(0)}$, which can be easily obtained by employing the first order continuous approximation of the optimizer. For convenience, we denote $\boldsymbol{\Gamma}_k := \boldsymbol{\gamma}_k^{(0)}$. Using the definition of $\dot{\boldsymbol{\theta}}$ in conjunction with Eq. (1) implies that

$$\dot{\boldsymbol{\theta}} = \lim_{\eta\to 0} \frac{\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k}{\eta} + \mathcal{O}(\eta^2) = -\lim_{\eta\to 0} \boldsymbol{U}_k(\nabla L; \boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_k) + \mathcal{O}(\eta^2). \tag{21}$$

Compared to Eq. (6), the lowest order of $\mathscr{U}_k$ needs to satisfy that

$$\lim_{\eta\to 0} \boldsymbol{\Gamma}_k = \lim_{\eta\to 0} \boldsymbol{U}_k(\nabla L; \boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_k). \tag{22}$$

Let the remainder terms of $\mathscr{U}_k$ in Eq. (6) be

$$\boldsymbol{\gamma}_k := \sum_{\sigma=1}^{+\infty} \eta^{\sigma-1} \boldsymbol{\gamma}_k^{(\sigma)}, \tag{23}$$

then the ODE Eq. (6) can be written as two distinct parts:

$$\dot{\boldsymbol{\theta}}(t) = -\boldsymbol{\Gamma}_k - \eta \boldsymbol{\gamma}_k. \tag{24}$$

Notice the power of η in Eq. (23) and the coefficient of $\boldsymbol{\gamma}_k$ in Eq. (24), since the power of η in the residual term of $\mathscr{U}_k$ is always greater than or equal to 1, there is at least one common η that can be extracted from the summation symbols, which makes a clear distinction between $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\gamma}_k$ in the form of Eq. (24), indicating that $\boldsymbol{\Gamma}_k$ is the lowest order, while $\boldsymbol{\gamma}_k$ is its higher-order remainder, i.e., counter terms.

So far we have only done some formal analysis, and have not yet obtained a concrete expression for $\boldsymbol{\gamma}_k$ and a method for solving it. Therefore, the problem to be solved now is how to obtain $\boldsymbol{\gamma}_k$ such that the discretization error of this continuous approximation meets the requirement of $\mathcal{O}(\eta^\rho)$ when $\boldsymbol{\Gamma}_k$ is given, and the solution to this problem is embodied in Theorem 1. Before giving the theorem, we first introduce the key ideas in the derivation process.

Obtain equalities from requiring $\varepsilon_k = \mathcal{O}(\eta^\rho)$ In order to control the discretization error ε_k for the k-th iteration, we associate ε_k with $\varepsilon_{k-1}, \varepsilon_{k-2}, \ldots$, until ε_0. As $\boldsymbol{\theta}_0$ is the initial value, it is guaranteed that $\varepsilon_0 = 0$. Thus, as long as the error in this process is bounded, the bound of ε_k would be controlled. To correlate the discretization errors of adjacent iterations, we perform a Taylor expansion of $\boldsymbol{\theta}(t_{k\pm1})$ at t_k, respectively. On the right-side interval $[t_k, t_{k+1})$, $\theta(t_{k+1})$ can be expanded as

$$\boldsymbol{\theta}(t_{k+1}) = \boldsymbol{\theta}(t_k) + \eta \dot{\boldsymbol{\theta}}^{+}(t_k) + \int_{t_k}^{t_{k+1}} \ddot{\boldsymbol{\theta}}^{+}(\tau)(t_{k+1} - \tau)\,\mathrm{d}\tau, \tag{25}$$

in which we use the superscript "+" to indicate that the definitions of $\dot{\boldsymbol{\theta}}(t)$ and $\ddot{\boldsymbol{\theta}}(t)$ on the right-side interval $[t_k, t_{k+1})$ should be used here, and the third term of Eq. (25) is the integral remainder of the Taylor expansion. Similarly, on the left-side interval $[t_{k-1}, t_k)$

$$\boldsymbol{\theta}(t_{k-1}) = \boldsymbol{\theta}(t_k) - \eta \dot{\boldsymbol{\theta}}^{-}(t_k) + \int_{t_k}^{t_{k-1}} \ddot{\boldsymbol{\theta}}^{-}(\tau)(t_{k-1} - \tau)\,\mathrm{d}\tau. \tag{26}$$

For convenience, we now unify the integral remainder in Eq. (25) and (26) as $\eta^2 \mathbf{I}_k^{\pm} = \int_{t_k}^{t_{k\pm1}} \ddot{\boldsymbol{\theta}}^{\pm}(\tau)(t_{k\pm1} - \tau)\,\mathrm{d}\tau$. Combined with the expression for $\dot{\boldsymbol{\theta}}$ on intervals $[t_k, t_{k+1})$ and $[t_{k-1}, t_k)$ (Eq. (24))

$$\dot{\boldsymbol{\theta}}^{+}(t) = -\boldsymbol{\Gamma}_k - \eta \boldsymbol{\gamma}_k, \quad \dot{\boldsymbol{\theta}}^{-}(t) = -\boldsymbol{\Gamma}_{k-1} - \eta \boldsymbol{\gamma}_{k-1}, \tag{27}$$

recalling the definition of the discretization error Eq. (2), we obtain a set of iteration relations for the discretization error ε_k

$$\begin{aligned} \varepsilon_{k+1} - \varepsilon_k &= (-\eta \boldsymbol{\Gamma}_k - \eta^2 \boldsymbol{\gamma}_k + \eta^2 \mathbf{I}_k^{+}) - (\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k), \\ \varepsilon_k - \varepsilon_{k-1} &= (-\eta \boldsymbol{\Gamma}_{k-1} - \eta^2 \boldsymbol{\gamma}_{k-1} - \eta^2 \mathbf{I}_k^{-}) - (\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}). \end{aligned} \tag{28}$$

If the update of two adjacent iterations can be related linearly as follows

$$\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k = \mu_k(\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}) - \eta \boldsymbol{f}_k(\mathbf{g}_k), \tag{29}$$

where $\eta f_k(\mathbf{g}_k)$ is the remainder term of the linearized relationship of updates between two adjacent iterations as a function of the local gradient, and since $\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k = \mathcal{O}(\eta)$, $\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1} = \mathcal{O}(\eta)$, there must be $\eta \boldsymbol{f}_k(\mathbf{g}_k) = \mathcal{O}(\eta)$, so we can always extract an η out[1]. Given Eq. (29), the difference between adjacent ε_k can be related as

$$\varepsilon_{k+1} - \varepsilon_k = \mu_k(\varepsilon_k - \varepsilon_{k-1}) - \eta(\boldsymbol{\Gamma}_k - \mu_k \boldsymbol{\Gamma}_{k-1} - \boldsymbol{f}_k(\mathbf{g}_k)) + \eta^2(\mathbf{I}_k^{+} + \mu_k \mathbf{I}_k^{-} - \boldsymbol{\gamma}_k + \mu_k \boldsymbol{\gamma}_{k-1}).$$

[1]Note that Eq. (29) is only an assumption to facilitate subsequent derivations. Some optimizers can satisfy this relation naturally, e.g. in the case of the HB method,

$$\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k = \beta(\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}) - \eta \mathbf{g}_k. \tag{30}$$

For the optimizers that do not directly satisfy Eq. (29) such as Adam, we propose a linearization strategy to apply, as we will elaborate in Section 3.

Suppose that

$$\boldsymbol{\Gamma}_k - \mu_k \boldsymbol{\Gamma}_{k-1} - \boldsymbol{f}_k(\mathbf{g}_k) = \mathcal{O}(\eta^\rho), \tag{31}$$

$$\mathbf{I}_k^+ + \mu_k \mathbf{I}_k^- - \boldsymbol{\gamma}_k + \mu_k \boldsymbol{\gamma}_{k-1} = \mathcal{O}(\eta^{\rho-1}), \tag{32}$$

then there will always be

$$\varepsilon_{k+1} - \varepsilon_k = \mu_k(\varepsilon_k - \varepsilon_{k-1}) + \mathcal{O}(\eta^{\rho+1}) \tag{33}$$

with the initial value $\boldsymbol{\theta}(0) = \boldsymbol{\theta}_0$ and $\varepsilon_0 = 0$.

Solve the equalities for $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\gamma}_k$ In Appendix A.2, we will show that the specific $\boldsymbol{\gamma}_k$ satisfying Eq. (32) can be obtained from $\boldsymbol{\Gamma}_k$. Hence we only need to find $\boldsymbol{\Gamma}_k$ that satisfies Eq. (22) and (31), since Eq. (32) is satisfied automatically. As a result, in the situation of finite time optimization (the iteration count k satisfies that $1 \leq k \leq N$ such that $t_N = N\eta = \mathcal{O}(1)$), the ODE Eq. (6) is an $\mathcal{O}(\eta^\rho)$-close continuous approximation for the discrete update Eq. (1) since

$$\varepsilon_k = \sum_{j=0}^{k-1} (\varepsilon_{j+1} - \varepsilon_j) + \varepsilon_0 = \mathcal{O}(\eta^\rho). \tag{34}$$

Thus we obtain a method of controlling the discretization error of the continuous approximation to $\mathcal{O}(\eta^\rho)$.

A.2 Proof of Theorem 1

Given Eq. (24) with unknown $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\gamma}_k$ that needs to be determined to give $\varepsilon_k = \mathcal{O}(\eta^\rho)$, our proof strategy has the following key steps:

1. derive the Taylor expansion of $\boldsymbol{\theta}(t_{k\pm1})$ to the first order of η with the remainder term in the integral form, which depends on the unknown $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\gamma}_k$ that needs to be determined later;
2. based on the Taylor expansion, derive expressions of ε_k;
3. require the discretization error ε_k to satisfy $\varepsilon_{k+1} - \varepsilon_k = \mathcal{O}(\eta^\rho)$ (Eq. (36)) for arbitrary $\rho \in \mathbb{Z}^+$, which is sufficient for us to guarantee $\varepsilon_k = \mathcal{O}(\eta^\rho)$ (Lemma 1), then we obtain a set of equalities for $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\gamma}_k$ (Lemma 2);
4. solve these equalities will give us the final formulations of $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\gamma}_k$.

Assuming that there exists a parametric curve $\boldsymbol{\theta}(t)$ on the interval $t \in [0, N\eta]$ as a continuous approximation of the N-step iterative process. The discretization error is defined as in Section 2.

A Taylor expansion of $\boldsymbol{\theta}(t)$ could be made in the neighborhood of t_k

$$\boldsymbol{\theta}(t_{k+1}) = \boldsymbol{\theta}(t_k + \eta) = \boldsymbol{\theta}(t_k) + \eta\dot{\boldsymbol{\theta}}(t_k^+) + \eta^2 \mathbf{I}_k^+ = \boldsymbol{\theta}(t_k) - \eta\boldsymbol{\Gamma}_k(\boldsymbol{\theta}_k) - \eta^2\boldsymbol{\gamma}_k(\boldsymbol{\theta}_k) + \eta^2 \mathbf{I}_k^+(\boldsymbol{\theta}_k),$$

$$\boldsymbol{\theta}(t_{k-1}) = \boldsymbol{\theta}(t_k - \eta) = \boldsymbol{\theta}(t_k) - \eta\dot{\boldsymbol{\theta}}(t_k^-) + \eta^2 \mathbf{I}_k^- = \boldsymbol{\theta}(t_k) + \eta\boldsymbol{\Gamma}_{k-1}(\boldsymbol{\theta}_k) + \eta^2\boldsymbol{\gamma}_{k-1}(\boldsymbol{\theta}_k) + \eta^2 \mathbf{I}_k^-(\boldsymbol{\theta}_k).$$

Thus

$$\boldsymbol{\theta}(t_{k+1}) - \boldsymbol{\theta}(t_k) = -\eta\boldsymbol{\Gamma}_k - \eta^2\boldsymbol{\gamma}_k + \eta^2 \mathbf{I}_k^+,\ \boldsymbol{\theta}(t_k) - \boldsymbol{\theta}(t_{k-1}) = -\eta\boldsymbol{\Gamma}_{k-1} - \eta^2\boldsymbol{\gamma}_{k-1} - \eta^2 \mathbf{I}_k^-.$$

On the other hand, the difference between neighboring ε_k can be written as

$$\varepsilon_{k+1} - \varepsilon_k = (\boldsymbol{\theta}(t_{k+1}) - \boldsymbol{\theta}(t_k)) - (\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k) = (-\eta\boldsymbol{\Gamma}_k - \eta^2\boldsymbol{\gamma}_k + \eta^2 \mathbf{I}_k^+) - (\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k),$$

$$\varepsilon_k - \varepsilon_{k-1} = (\boldsymbol{\theta}(t_k) - \boldsymbol{\theta}(t_{k-1})) - (\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}) = (-\eta\boldsymbol{\Gamma}_{k-1} - \eta^2\boldsymbol{\gamma}_{k-1} - \eta^2 \mathbf{I}_k^-) - (\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}).$$

If the update of two neighboring iterations can be related linearly as follows

$$\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k = \mu_k(\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}) - \eta \boldsymbol{f}_k(\mathbf{g}_k), \tag{35}$$

then the difference between neighboring ε_k can also be related as

$$\begin{aligned}\varepsilon_{k+1} - \varepsilon_k &= (-\eta\boldsymbol{\Gamma}_k - \eta^2\boldsymbol{\gamma}_k + \eta^2 \mathbf{I}_k^+) - \left\{\mu_k \left[-\eta\boldsymbol{\Gamma}_{k-1} - \eta^2\boldsymbol{\gamma}_{k-1} - \eta^2 \mathbf{I}_k^- - (\varepsilon_k - \varepsilon_{k-1})\right] - \eta \boldsymbol{f}_k(\mathbf{g}_k)\right\} \\ &= \mu_k(\varepsilon_k - \varepsilon_{k-1}) - \eta\left(\boldsymbol{\Gamma}_k - \mu_k\boldsymbol{\Gamma}_{k-1} - \boldsymbol{f}_k(\mathbf{g}_k)\right) + \eta^2(\mathbf{I}_k^+ + \mu_k \mathbf{I}_k^- - \boldsymbol{\gamma}_k + \mu_k\boldsymbol{\gamma}_{k-1}).\end{aligned} \tag{36}$$

We need to correlate the order of the difference of neighboring ε_k with the order of the final ε_N to obtain the conditions under which $\varepsilon_N = \mathcal{O}(\eta^\rho)$ holds. Here we first give a useful lemma.

Lemma 1 (Propagation of Discretization Errors)**.** *For the parameter sequence $\boldsymbol{\theta}_k$ obtained by the optimizer and the segmented continuous approximation $\boldsymbol{\theta}(t)$ defined on $[0, T]$, $T = N\eta$, the discretization error is defined as in Eq.* (2)*. If T is fixed and $\varepsilon_{k+1} - \varepsilon_k = \mathcal{O}(\eta^{\rho+1})$ holds for any $k \in [N]$ ($\varepsilon_0 = 0$), then $\varepsilon_k = \mathcal{O}(\eta^\rho)$.*

Proof. Notice that

$$\begin{aligned}\varepsilon_k =&(\varepsilon_k - \varepsilon_{k-1}) + (\varepsilon_{k-1} - \varepsilon_{k-2}) + ... + (\varepsilon_1 - \varepsilon_0) + \varepsilon_0\\ =&k \times \mathcal{O}(\eta^{\rho+1}) \leq N \times \mathcal{O}(\eta^{\rho+1}) = \frac{T}{\eta} \times \mathcal{O}(\eta^{\rho+1}) = \mathcal{O}(\eta^\rho).\end{aligned}$$

The proposition is proved. □

Lemma 1 states that $\varepsilon_{k+1} - \varepsilon_k = \mathcal{O}(\eta^{\rho+1})$ is a sufficient condition for $\varepsilon_N = \mathcal{O}(\eta^\rho)$. Therefore, we can first make $\varepsilon_{k+1} - \varepsilon_k = \mathcal{O}(\eta^{\rho+1})$ hold. So we then give a sufficient condition for $\varepsilon_{k+1} - \varepsilon_k = \mathcal{O}(\eta^{\rho+1})$.

Lemma 2 (Sufficient conditions for the controlled bound of discretization error)**.** *If*

$$\boldsymbol{\Gamma}_k - \mu_k \boldsymbol{\Gamma}_{k-1} - \boldsymbol{f}_k(\mathbf{g}_k) =\mathcal{O}(\eta^\rho), \tag{37}$$

$$\mathbf{I}_k^+ + \mu_k \mathbf{I}_k^- - \boldsymbol{\gamma}_k + \mu_k \boldsymbol{\gamma}_{k-1} =\mathcal{O}(\eta^{\rho-1}) \tag{38}$$

hold for any $k \in [N]$, and the derivatives of each order of $\boldsymbol{f}_k$ are bounded, then

$$\varepsilon_{k+1} - \varepsilon_k = \mathcal{O}(\eta^{\rho+1}). \tag{39}$$

Proof. We use mathematical induction to prove it. It is known that the initial value $\varepsilon_0 = 0 = \mathcal{O}(\eta^{\rho+1})$. Suppose that $\varepsilon_k - \varepsilon_{k-1} = \mathcal{O}(\eta^{\rho+1})$, then from Eq. (36), we have

$$\varepsilon_{k+1} - \varepsilon_k = \mu_k \mathcal{O}(\eta^{\rho+1}) - \eta \mathcal{O}(\eta^\rho) + \eta^2 \mathcal{O}(\eta^{\rho-1}) = \mathcal{O}(\eta^{\rho+1}). \tag{40}$$

Thus the proposition is proved. □

For Eq. (37), since the derivatives of each order of the loss function L are bounded, there should be

$$\mathbf{g} - \mathbf{g}_k = \mathbf{g}(\boldsymbol{\theta}(t_k)) - \mathbf{g}(\boldsymbol{\theta}_k) = \mathcal{O}(\eta^\rho).$$

Similarly, since the derivatives of each order of $\boldsymbol{f}_k$ are bounded, there should also be

$$\boldsymbol{f}_k(\mathbf{g}) - \boldsymbol{f}_k(\mathbf{g}_k) = \mathcal{O}(\eta^\rho).$$

Thus, we only need to get $\boldsymbol{\Gamma}_k$ according to the recursion

$$\boldsymbol{\Gamma}_k = \mu_k \boldsymbol{\Gamma}_{k-1} + \boldsymbol{f}_k(\mathbf{g}) \tag{41}$$

to make $\boldsymbol{\Gamma}_k - \mu_k \boldsymbol{\Gamma}_{k-1} - \boldsymbol{f}_k(\mathbf{g}_k) = \mathcal{O}(\eta^\rho)$.

For Eq. (38), we take a different line of thinking. The definition of $\boldsymbol{\gamma}_k$ is an infinite series with respect to η, whereas in order to fulfill the requirements of Eq. (38), we do not actually need so many terms. In fact, we can truncate it as

$$\boldsymbol{\gamma}_k = \sum_{\sigma=1}^{+\infty} \eta^{\sigma-1} \boldsymbol{\gamma}_k^{(\sigma)} = \sum_{\sigma=1}^{\rho-1} \eta^{\sigma-1} \boldsymbol{\gamma}_k^{(\sigma)} + \sum_{\sigma=\rho}^{+\infty} \eta^{\sigma-1} \boldsymbol{\gamma}_k^{(\sigma)} = \sum_{\sigma=1}^{\rho-1} \eta^{\sigma-1} \boldsymbol{\gamma}_k^{(\sigma)} + \mathcal{O}(\eta^{\rho-1}).$$

If $\mathbf{I}_k^+$ and $\mathbf{I}_k^-$ can also be written as infinite series of η separately and then truncated to $\mathcal{O}(\eta^{\rho-1})$, we are assured that Eq. (37) holds. Next, we will try to write $\mathbf{I}_k^+$ and $\mathbf{I}_k^-$ in the series form of η and give an expression for $\boldsymbol{\gamma}_k$.

Write $\mathbf{I}_k^+$ and $\mathbf{I}_k^-$ in the form of the integral remainder term of the Taylor expansion

$$\mathbf{I}_k^\pm = \frac{1}{\eta^2} \int_{k\eta}^{k\eta \pm \eta} \left[\ddot{\boldsymbol{\theta}}^\pm(\tau)(k\eta \pm \eta - \tau) \right] \mathrm{d}\tau = \frac{1}{\eta^2} \int_0^{\pm\eta} \left[\sum_{i=0}^{+\infty} \frac{1}{i!} \frac{\mathrm{d}^i \ddot{\boldsymbol{\theta}}^\pm(t_k)}{\mathrm{d}t^i} \tau^i \right] (\pm\eta - \tau)\, \mathrm{d}\tau$$

$$= \frac{1}{\eta^2}\left[\sum_{i=0}^{+\infty}\frac{1}{i!}\frac{\mathrm{d}^{i+2}\boldsymbol{\theta}^{\pm}(t_k)}{\mathrm{d}t^{i+2}}\int_0^{\pm\eta}\tau^i(\pm\eta-\tau)\,\mathrm{d}\tau\right] = \sum_{i=0}^{+\infty}\frac{(\pm\eta)^i}{(i+2)!}\frac{\mathrm{d}^{i+2}\boldsymbol{\theta}^{\pm}(t_k)}{\mathrm{d}t^{i+2}}, \tag{42}$$

where $\frac{\mathrm{d}^i\boldsymbol{\theta}^+(t_k)}{\mathrm{d}t^i}$ can continue to be expanded using the chain rule for derivatives

$$\begin{aligned}\frac{\mathrm{d}^i\boldsymbol{\theta}^+(t_k)}{\mathrm{d}t^i} &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}^{i-1}\boldsymbol{\theta}^+(t_k)}{\mathrm{d}t^{i-1}}\right) = \dot{\boldsymbol{\theta}}^+(t_k)\cdot\nabla\left(\frac{\mathrm{d}^{i-1}\boldsymbol{\theta}^+(t_k)}{\mathrm{d}t^{i-1}}\right)\\ &= -(\boldsymbol{\Gamma}_k+\eta\boldsymbol{\gamma}_k)\cdot\nabla\left(\frac{\mathrm{d}^{i-1}\boldsymbol{\theta}^+(t_k)}{\mathrm{d}t^{i-1}}\right) = (-1)^i(\mathbf{L}_k)^{i-1}(\boldsymbol{\Gamma}_k+\eta\boldsymbol{\gamma}_k),\end{aligned} \tag{43}$$

where $\mathbf{L}_k$ is a differential operator defined as $\mathbf{L}_k = (\boldsymbol{\Gamma}_k+\eta\boldsymbol{\gamma}_k)\cdot\nabla$.

Similarly,

$$\frac{\mathrm{d}^i\boldsymbol{\theta}^-(t_k)}{\mathrm{d}t^i} = (-1)^i(\mathbf{L}_{k-1})^{i-1}(\boldsymbol{\Gamma}_{k-1}+\eta\boldsymbol{\gamma}_{k-1}). \tag{44}$$

Then we can rewrite $\frac{\mathrm{d}^i\boldsymbol{\theta}^+(t_k)}{\mathrm{d}t^i}$ as

$$\begin{aligned}\frac{\mathrm{d}^i\boldsymbol{\theta}^+(t_k)}{\mathrm{d}t^i} =&(-1)^i\left(\sum_{\sigma_1=0}^{+\infty}\eta^{\sigma_1}\boldsymbol{\gamma}_k^{(\sigma_1)}\cdot\nabla\right)\left(\sum_{\sigma_2=0}^{+\infty}\eta^{\sigma_2}\boldsymbol{\gamma}_k^{(\sigma_2)}\cdot\nabla\right)\cdots\\ &\cdots\left(\sum_{\sigma_{i-1}=0}^{+\infty}\eta^{\sigma_{i-1}}\boldsymbol{\gamma}_k^{(\sigma_{i-1})}\cdot\nabla\right)\left(\sum_{\sigma_i=0}^{+\infty}\eta^{\sigma_i}\boldsymbol{\gamma}_k^{(\sigma_i)}\right)\end{aligned} \tag{45}$$

$$=(-1)^i\sum_{\sigma_1=0}^{+\infty}\sum_{\sigma_2=0}^{+\infty}\cdots\sum_{\sigma_i=0}^{+\infty}\eta^{\left(\sum_{j=1}^{i}\sigma_j\right)}\mathbf{L}_k^{(\sigma_1)}\mathbf{L}_k^{(\sigma_2)}\cdots\mathbf{L}_k^{(\sigma_{i-1})}\boldsymbol{\gamma}_k^{(\sigma_i)}, \tag{46}$$

in which $\mathbf{L}_k^{(\sigma_j)} = \boldsymbol{\gamma}_k^{(\sigma_j)}\cdot\nabla$.

Substituting the above equation back into the expression for $\mathbf{I}_k^+$ yields

$$\begin{aligned}\mathbf{I}_k^+ =&\sum_{i=0}^{+\infty}\frac{\eta^i}{(i+2)!}\frac{\mathrm{d}^{i+2}\boldsymbol{\theta}^{\pm}(t_k)}{\mathrm{d}t^{i+2}}\\ =&\sum_{i=0}^{+\infty}\sum_{\sigma_1=0}^{+\infty}\sum_{\sigma_2=0}^{+\infty}\cdots\sum_{\sigma_{i+2}=0}^{+\infty}\frac{(-1)^{i+2}\eta^{\left(i+\sum_{j=1}^{i+2}\sigma_j\right)}}{(i+2)!}\mathbf{L}_k^{(\sigma_1)}\mathbf{L}_k^{(\sigma_2)}\cdots\mathbf{L}_k^{(\sigma_{i+1})}\boldsymbol{\gamma}_k^{(\sigma_{i+2})}\\ =&\sum_{i=0}^{+\infty}\sum_{m=0}^{+\infty}\sum_{\sum_{j=1}^{i+2}\sigma_j=m}\frac{(-1)^{i+2}\eta^{i+m}}{(i+2)!}\mathbf{L}_k^{(\sigma_1)}\mathbf{L}_k^{(\sigma_2)}\cdots\mathbf{L}_k^{(\sigma_{i+1})}\boldsymbol{\gamma}_k^{(\sigma_{i+2})}\\ =&\sum_{p=0}^{+\infty}\sum_{q=2}^{p+2}\sum_{\sum_{j=1}^{q}\sigma_j=p-q+2}\frac{(-1)^q\eta^p}{q!}\mathbf{L}_k^{(\sigma_1)}\mathbf{L}_k^{(\sigma_2)}\cdots\mathbf{L}_k^{(\sigma_{q-1})}\boldsymbol{\gamma}_k^{(\sigma_q)}.\end{aligned} \tag{47}$$

Similarly, for $\mathbf{I}_k^-$, we have

$$\begin{aligned}\mathbf{I}_k^- =&\sum_{i=0}^{+\infty}\frac{(-\eta)^i}{(i+2)!}\frac{\mathrm{d}^{i+2}\boldsymbol{\theta}^-(t_k)}{\mathrm{d}t^{i+2}}\\ =&\sum_{i=0}^{+\infty}\sum_{\sigma_1=0}^{+\infty}\sum_{\sigma_2=0}^{+\infty}\cdots\sum_{\sigma_{i+2}=0}^{+\infty}\frac{(-1)^{2i+2}\eta^{\left(i+\sum_{j=1}^{i+2}\sigma_j\right)}}{(i+2)!}\mathbf{L}_{k-1}^{(\sigma_1)}\mathbf{L}_{k-1}^{(\sigma_2)}\cdots\mathbf{L}_{k-1}^{(\sigma_{i+1})}\boldsymbol{\gamma}_{k-1}^{(\sigma_{i+2})}\\ =&\sum_{p=0}^{+\infty}\sum_{q=2}^{p+2}\sum_{\sum_{j=1}^{q}\sigma_j=p-q+2}\frac{\eta^p}{q!}\mathbf{L}_{k-1}^{(\sigma_1)}\mathbf{L}_{k-1}^{(\sigma_2)}\cdots\mathbf{L}_{k-1}^{(\sigma_{q-1})}\boldsymbol{\gamma}_{k-1}^{(\sigma_q)}.\end{aligned} \tag{48}$$

Thus, synthesizing the series expansions of $\boldsymbol{\gamma}_k$ and $\mathbf{I}_k^{\pm}$, we get the expressions for the coefficients of each terms of $\boldsymbol{\gamma}_k$ when $\sigma \geq 1$

$$\left(\sum_{q=2}^{\sigma+1}\sum_{\sum_{j=1}^{q}\sigma_j=\sigma-q+1}\frac{(-1)^q}{q!}\mathbf{L}_k^{(\sigma_1)}\mathbf{L}_k^{(\sigma_2)}\cdots\mathbf{L}_k^{(\sigma_{q-1})}\boldsymbol{\gamma}_k^{(\sigma_q)}\right.$$
$$\left.+\mu_k\sum_{q=2}^{\sigma+1}\sum_{\sum_{j=1}^{q}\sigma_j=\sigma-q+1}\frac{1}{q!}\mathbf{L}_{k-1}^{(\sigma_1)}\mathbf{L}_{k-1}^{(\sigma_2)}\cdots\mathbf{L}_{k-1}^{(\sigma_{q-1})}\boldsymbol{\gamma}_{k-1}^{(\sigma_q)}-\boldsymbol{\gamma}_k^{(\sigma)}+\mu_k\boldsymbol{\gamma}_{k-1}^{(\sigma)}\right)\eta^{\sigma-1}=0. \quad (49)$$

Denote that

$$\boldsymbol{\chi}_k^{(\sigma)}=\sum_{q=2}^{\sigma+1}\sum_{\sum_{j=1}^{q}\sigma_j=\sigma-q+1}\frac{1}{q!}\left[(-1)^q\mathbf{L}_k^{(\sigma_1)}\mathbf{L}_k^{(\sigma_2)}\cdots\mathbf{L}_k^{(\sigma_{q-1})}\boldsymbol{\gamma}_k^{(\sigma_q)}.+\mu_k\mathbf{L}_{k-1}^{(\sigma_1)}\mathbf{L}_{k-1}^{(\sigma_2)}\cdots\mathbf{L}_{k-1}^{(\sigma_{q-1})}\boldsymbol{\gamma}_{k-1}^{(\sigma_q)}\right],$$

then $\boldsymbol{\gamma}_k^{(\sigma)}$ can be expressed as

$$\boldsymbol{\gamma}_k^{(\sigma)}=\mu_k\boldsymbol{\gamma}_{k-1}^{(\sigma)}+\boldsymbol{\chi}_k^{(\sigma)}=\sum_{j=0}^{k}\left(\prod_{i=j+1}^{k}\mu_i\right)\boldsymbol{\chi}_j^{(\sigma)}. \quad (50)$$

As a result, the $\boldsymbol{\gamma}_k=\sum_{\sigma=1}^{\rho-1}\eta^{\rho-1}\boldsymbol{\gamma}_k^{(\sigma)}$ thus obtained can satisfy Eq. (38), so the conditions of Lemma 2 are all satisfied, eventually Theorem 1 is proved.

A.3 Continuous Approximations for GD and HB

We now apply the unified framework in Theorem 1 to Gradient Descent (GD) and Heavy-Ball (HB) momentum method. GD, the simplest gradient-based optimizer, updates the parameter along the negative gradient direction

$$\boldsymbol{\theta}_{k+1}=\boldsymbol{\theta}_k-\eta\mathbf{g}_k, \quad (51)$$

which can be covered by our requirement for the update of adjacent iterations Eq. (29) if

$$\mu_k=0, \quad \boldsymbol{f}_k(\mathbf{g}_k)=\mathbf{g}_k$$

and $\boldsymbol{\Gamma}_k$ should satisfy

$$\boldsymbol{\Gamma}_k=\mu_k\boldsymbol{\Gamma}_{k-1}+\boldsymbol{f}_k(\mathbf{g})=\mathbf{g}. \quad (52)$$

An important variant of GD is Heavy-Ball (HB) momentum method, which follows a two-step manner to update the parameter

$$\textbf{Discrete HB}\ \begin{cases}\mathbf{m}_{k+1}=\beta\mathbf{m}_k+\mathbf{g}\\ \boldsymbol{\theta}_{k+1}=\boldsymbol{\theta}_k-\eta\mathbf{m}_{k+1}\end{cases}. \quad (53)$$

Alternatively, this two-step scheme can also be directly expressed as the form of one-step update in Eq. (29)

$$\boldsymbol{\theta}_{k+1}=\boldsymbol{\theta}_k-\eta\mathbf{g}_k+\beta(\boldsymbol{\theta}_k-\boldsymbol{\theta}_{k-1}) \quad (54)$$

by letting $\mu_k=\beta$ and $\boldsymbol{f}_k(\mathbf{g}_k)=\mathbf{g}_k$, while $\boldsymbol{\Gamma}_k$ should satisfy

$$\boldsymbol{\Gamma}_k=\mu_k\boldsymbol{\Gamma}_{k-1}+\boldsymbol{f}_k(\mathbf{g})=\frac{1-\beta^{k+1}}{1-\beta}\mathbf{g}. \quad (55)$$

Kovachki & Stuart (2021) has given the first order continuous approximation of HB as

$$\dot{\boldsymbol{\theta}}=-\frac{1}{1-\beta}\nabla L. \quad (56)$$

Compared to $\boldsymbol{\Gamma}_k$, we can find that when $t_k=k\eta$ is fixed,

$$\lim_{\eta\to 0}\boldsymbol{\Gamma}_k=\lim_{k\to+\infty}\boldsymbol{\Gamma}_k=\frac{1}{1-\beta}\nabla L, \quad (57)$$

which indicates the reasonability of our $\boldsymbol{\Gamma}_k$.

When $\beta=0$, the HB momentum method would degenerate to GD. Therefore, below we discuss the HB momentum method, and generalize to the case for GD by simply setting $\beta=0$.

Corollary 1.3 (Arbitrary Order Continuous Approximation of Heavy-Ball Method)**.** *Following the framework described in Theorem 1, the arbitrary order continuous approximation of the Heavy-Ball method Eq.* (53) *can be obtained recursively (assuming that k is large enough)*

$$\boldsymbol{\gamma}_k^{(0)} = \frac{1}{1-\beta}\mathbf{g}, \tag{58}$$

$$\boldsymbol{\gamma}_k^{(1)} = \frac{1+\beta}{2(1-\beta)^3}\mathbf{g}\cdot\nabla\mathbf{g}, \tag{59}$$

$$\boldsymbol{\gamma}_k^{(2)} = \frac{(1+\beta)^2}{4(1-\beta)^5}\left[\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \frac{1+10\beta+\beta^2}{3(1+\beta)^2}\mathbf{g}\cdot\nabla(\mathbf{g}\cdot\nabla\mathbf{g})\right], \tag{60}$$

$$\vdots$$

$$\boldsymbol{\gamma}_k^{(\sigma)} = \sum_{j=0}^{k}\beta^{k-j}\boldsymbol{\chi}_j^{(\sigma)}, \tag{61}$$

where $\chi_j^{(\sigma)}$ is defined in Eq. (10)*.*

Corollary 1.3 provides a method for obtaining higher order continuous approximations for HB. Preserving $\gamma_k^{(\sigma)}$ up to the order $\sigma = 2$ such that $\gamma_k = \sum_{\sigma=0}^{2}\gamma_k^{(\sigma)}$, we obtain the third order continuous approximation of HB as

$$\dot{\boldsymbol{\theta}} = -\frac{\mathbf{g}}{1-\beta} - \frac{\eta(1+\beta)}{2(1-\beta)^3}\mathbf{g}\cdot\nabla\mathbf{g} - \frac{\eta^2(1+\beta)^2}{4(1-\beta)^5}\left[\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \zeta(\beta)\mathbf{g}\cdot\nabla(\mathbf{g}\cdot\nabla\mathbf{g})\right], \tag{62}$$

where $\zeta(\beta)$ is defined as in Eq. (**??**). By comparison, the second order continuous approximation obtained by omitting the last term of Eq. (62) is in agreement with that given by Ghosh et al. (2023), and the third order term also coincides with that proposed by Lyu et al. (2025), which further validates our method. By taking $\beta = 0$, our results can be easily generalized to give the continuous approximation for GD, as shown below.

Corollary 1.4 (Arbitrary Order Continuous Approximation of Gradient Descent)**.** *Following the framework in Theorem 1, the arbitrary order continuous approximation for GD (Eq.* (51)*) can be obtained recursively*

$$\boldsymbol{\gamma}_k^{(0)} = \mathbf{g}, \tag{63}$$

$$\boldsymbol{\gamma}_k^{(1)} = \frac{1}{2}\mathbf{g}\cdot\nabla\mathbf{g}, \tag{64}$$

$$\boldsymbol{\gamma}_k^{(2)} = \frac{1}{4}\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \frac{1}{12}\mathbf{g}\cdot\nabla(\mathbf{g}\cdot\nabla\mathbf{g}), \tag{65}$$

$$\vdots$$

$$\boldsymbol{\gamma}_k^{(\sigma)} = \chi_k^{(\sigma)}, \tag{66}$$

where $\chi_k^{(\sigma)}$ is defined in Eq. (10)*.*

Similarly, let $\gamma_k = \sum_{\sigma=0}^{2}\gamma_k^{(\sigma)}$, the third-order continuous approximation of GD could be written as

$$\dot{\boldsymbol{\theta}} = -\mathbf{g} - \frac{\eta}{2}\mathbf{g}\cdot\nabla\mathbf{g} - \frac{\eta^2}{4}\left[\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \frac{1}{3}\mathbf{g}\cdot\nabla(\mathbf{g}\cdot\nabla\mathbf{g})\right], \tag{67}$$

which is the same as the conclusion of Miyagawa (2023). Comparison with the result of Rosca et al. (2023) shows that our continuous approximation matches that obtained by backward error analysis, thus the analysis process and conclusion of the implicit bias are also consistent and will not be repeated here. Following this procedure, we can obtain an arbitrary order continuous approximation of GD, which would provide an effective contribution to a more thorough understanding of GD.

A.3.1 $\rho = 2$.

Proof. According to Theorem 1, we need to obtain $\boldsymbol{\gamma}_k^{(1)}$ to get a continuous approximation of HB (and GD) with discretization error to $\mathcal{O}(\eta^2)$. It is known that $\boldsymbol{\gamma}_k^{(1)}$ depends on $\boldsymbol{\chi}_j^{(1)}$, and

$$\boldsymbol{\chi}_j^{(1)} = \sum_{\mathcal{S}_0^2} \frac{1}{2} \left[\mathbf{L}_j^{(\sigma_1)} \boldsymbol{\gamma}_j^{(\sigma_2)} + \mu_j \mathbf{L}_{j-1}^{(\sigma_1)} \boldsymbol{\gamma}_{j-1}^{(\sigma_2)} \right], \tag{68}$$

where the set $\mathcal{S}$ on which the summation sign acts is $\mathcal{S}_0^2 = \{(\sigma_1 = 0, \sigma_2 = 0)\}$.

Together with our known first order continuous approximation for HB

$$\boldsymbol{\Gamma}_k = \boldsymbol{\gamma}_k^{(0)} = \frac{1 - \beta^{k+1}}{1 - \beta} \mathbf{g} \approx \frac{1}{1 - \beta} \mathbf{g}, \tag{69}$$

here we are mainly concerned with the continuous approximation when k is large enough, so we simply ignore β^{k+1}. Thus we get

$$\chi_j^{(1)} = \frac{1}{2} \left[\boldsymbol{\Gamma}_j \cdot \nabla \boldsymbol{\Gamma}_j + \beta \boldsymbol{\Gamma}_{j-1} \cdot \nabla \boldsymbol{\Gamma}_{j-1} \right] \approx \frac{1 + \beta}{2(1 - \beta)^2} \mathbf{g} \cdot \nabla \mathbf{g}. \tag{70}$$

Therefore, it is obtained that

$$\boldsymbol{\gamma}_k^{(1)} = \sum_{j=0}^{k} \beta^{k-j} \boldsymbol{\chi}_j^{(1)} \approx \frac{1 + \beta}{2(1 - \beta)^3} \mathbf{g} \cdot \nabla \mathbf{g}. \tag{71}$$

The continuous approximation of GD with discretization error to $\mathcal{O}(\eta^2)$ can be obtained simply by taking $\beta = 0$ in Eq. (71) $\boldsymbol{\gamma}_k^{(1)} = \frac{1}{2} \mathbf{g} \cdot \nabla \mathbf{g}$. Thus we get $\boldsymbol{\gamma}_k^{(1)}$ in Corollary 1.4 and Corollary 1.3.

A.3.2 $\rho = 3$.

Proof. In this case, we take $\rho = 3$ in Theorem 1, therefore we should obtain $\boldsymbol{\gamma}_k^{(1)}$ and $\boldsymbol{\gamma}_k^{(2)}$ to get third order continuous approximations for GD and HB. $\boldsymbol{\gamma}_k^{(1)}$ is already obtained in the case $\rho = 2$, here we focus on the $\boldsymbol{\gamma}_k^{(2)}$.

Similarly, we first need to obtain $\boldsymbol{\chi}_j^{(2)}$. Here from Theorem 1 we have

$$\boldsymbol{\chi}_j^{(2)} = \sum_{\mathcal{S}_1^2} \frac{1}{2} \left[\mathbf{L}_j^{(\sigma_1)} \boldsymbol{\gamma}_j^{(\sigma_2)} + \mu_j \mathbf{L}_{j-1}^{(\sigma_1)} \boldsymbol{\gamma}_{j-1}^{(\sigma_2)} \right] + \sum_{\mathcal{S}_0^3} \frac{1}{6} \left[-\mathbf{L}_j^{(\sigma_1)} \mathbf{L}_j^{(\sigma_2)} \boldsymbol{\gamma}_j^{(\sigma_3)} + \mu_j \mathbf{L}_{j-1}^{(\sigma_1)} \mathbf{L}_{j-1}^{(\sigma_2)} \boldsymbol{\gamma}_{j-1}^{(\sigma_3)} \right],$$

where the sets $\mathcal{S}$ on which the summation signs act are

$$\begin{aligned} \mathcal{S}_1^2 =& \{(\sigma_1 = 1, \sigma_2 = 0), (\sigma_1 = 0, \sigma_2 = 1)\}, \\ \mathcal{S}_0^3 =& \{(\sigma_1 = 0, \sigma_2 = 0, \sigma_3 = 0)\}. \end{aligned}$$

Thus there are three terms in $\boldsymbol{\chi}_j^{(2)}$,

$$\boldsymbol{\chi}_j^{(2)} = D_1 + D_2 + D_3, \tag{72}$$

$$D_1 := \frac{1}{2} \left[\mathbf{L}_j^{(1)} \boldsymbol{\gamma}_j^{(0)} + \mu_j \mathbf{L}_{j-1}^{(1)} \boldsymbol{\gamma}_{j-1}^{(0)} \right], \tag{73}$$

$$D_2 := \frac{1}{2} \left[\mathbf{L}_j^{(0)} \boldsymbol{\gamma}_j^{(1)} + \mu_j \mathbf{L}_{j-1}^{(0)} \boldsymbol{\gamma}_{j-1}^{(1)} \right], \tag{74}$$

$$D_3 := \frac{1}{6} \left[-\mathbf{L}_j^{(0)} \mathbf{L}_j^{(0)} \boldsymbol{\gamma}_j^{(0)} + \mu_j \mathbf{L}_{j-1}^{(0)} \mathbf{L}_{j-1}^{(0)} \boldsymbol{\gamma}_{j-1}^{(0)} \right]. \tag{75}$$

Recall that $\boldsymbol{\gamma}_j^{(0)} = \boldsymbol{\Gamma}_j \approx \mathbf{g}/(1 - \beta)$, then reusing the previously obtained $\boldsymbol{\gamma}_k^{(1)}$ in Eq. (71), we have

$$D_1 \approx \frac{1 + \beta}{2} \left[\frac{(1 + \beta)}{2(1 - \beta)^3} \mathbf{g} \cdot \nabla \mathbf{g} \cdot \nabla \left(\frac{\mathbf{g}}{1 - \beta} \right) \right] = \frac{(1 + \beta)^2}{4(1 - \beta)^4} \mathbf{g} \cdot \nabla \mathbf{g} \cdot \nabla \mathbf{g}, \tag{76}$$

$$D_2 \approx \frac{1+\beta}{2}\left[\frac{\mathbf{g}}{1-\beta}\cdot\nabla\left(\frac{1+\beta}{2(1-\beta)^3}\mathbf{g}\cdot\nabla\mathbf{g}\right)\right] = \frac{(1+\beta)^2}{4(1-\beta)^4}\mathbf{g}\cdot\nabla\left(\mathbf{g}\cdot\nabla\mathbf{g}\right), \tag{77}$$

$$D_3 \approx -\frac{1-\beta}{6}\left\{\frac{\mathbf{g}}{1-\beta}\cdot\nabla\left[\frac{\mathbf{g}}{1-\beta}\cdot\nabla\left(\frac{\mathbf{g}}{1-\beta}\right)\right]\right\} = -\frac{1}{6(1-\beta)^2}\mathbf{g}\cdot\nabla\left(\mathbf{g}\cdot\nabla\mathbf{g}\right) \tag{78}$$

Thus there is

$$\begin{aligned}\boldsymbol{\chi}_j^{(2)} \approx& \frac{(1+\beta)^2}{4(1-\beta)^4}\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \frac{(1+\beta)^2}{4(1-\beta)^4}\mathbf{g}\cdot\nabla\left(\mathbf{g}\cdot\nabla\mathbf{g}\right) - \frac{1}{6(1-\beta)^2}\mathbf{g}\cdot\nabla\left(\mathbf{g}\cdot\nabla\mathbf{g}\right)\\ =& \frac{(1+\beta)^2}{4(1-\beta)^4}\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \frac{1+10\beta+\beta^2}{12(1-\beta)^4}\mathbf{g}\cdot\nabla\left(\mathbf{g}\cdot\nabla\mathbf{g}\right).\end{aligned}$$

Now we can obtain that

$$\boldsymbol{\gamma}_k^{(2)} = \sum_{j=0}^{k}\beta^{k-j}\boldsymbol{\chi}_j^{(2)} \approx \frac{(1+\beta)^2}{4(1-\beta)^5}\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \frac{1+10\beta+\beta^2}{12(1-\beta)^5}\mathbf{g}\cdot\nabla\left(\mathbf{g}\cdot\nabla\mathbf{g}\right). \tag{79}$$

When taking $\beta = 0$ in Eq. (79), we can obtain the continuous approximation of GD with discretization error to $\mathcal{O}(\eta^3)$ as

$$\boldsymbol{\gamma}_k^{(2)} \approx \frac{1}{4}\mathbf{g}\cdot\nabla\mathbf{g}\cdot\nabla\mathbf{g} + \frac{1}{12}\mathbf{g}\cdot\nabla\left(\mathbf{g}\cdot\nabla\mathbf{g}\right). \tag{80}$$

Thus we get $\boldsymbol{\gamma}_k^{(2)}$ in Corollary 1.4 and Corollary 1.3.

A.4 Proof of Corollary 1.1

The iterative progress for Adam is

$$\begin{cases}\mathbf{m}_{k+1} = \beta\mathbf{m}_k + (1-\beta)\mathbf{g}_k,\ \hat{\mathbf{m}}_{k+1} = \dfrac{\mathbf{m}_{k+1}}{1-\beta^{k+1}}\\ \mathbf{v}_{k+1} = \alpha\mathbf{v}_k + (1-\alpha)\mathbf{g}_k^2,\ \hat{\mathbf{v}}_{k+1} = \dfrac{\mathbf{v}_{k+1}}{1-\alpha^{k+1}}\\ \Delta_{k+1} = \boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k = -\eta\dfrac{\hat{\mathbf{m}}_{k+1}}{\sqrt{\hat{\mathbf{v}}_{k+1}+\varepsilon}} \overset{k\to+\infty}{\approx} -\eta\dfrac{\mathbf{m}_{k+1}}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}}\end{cases}. \tag{81}$$

It is difficult to construct a suitable linear relationship between Δ_{k+1} and Δ_k directly, because the relationship is affected by both the change in the numerator and the change in the denominator. However, we can consider a decoupling approach whereby changes in the numerator are considered separately from changes in the denominator. When their impacts are made clear separately, they could then be combined again.

As mentioned in Section 3, we can divide Δ_k into the following two parts to be considered separately:

$$\Delta_{1,k+1} = -\eta\frac{\mathbf{m}_{k+1}}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}} = -\eta\frac{\beta\mathbf{m}_k}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}} - \eta\frac{(1-\beta)\mathbf{g}_k}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}} = \beta\Delta_{1,k} - \eta\frac{(1-\beta)\mathbf{g}_k}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}},$$

thus $\mu_{1,k} = \beta$, $\boldsymbol{f}_{1,k}(\mathbf{g}_k) = \frac{(1-\beta)\mathbf{g}_k}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}}$, and

$$\begin{aligned}\Delta_{2,k+1} =& -\eta\frac{\mathbf{m}_k}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}} = -\eta\frac{\mathbf{m}_k}{\sqrt{\mathbf{v}_k + (1-\alpha)(\mathbf{g}_k^2 - \mathbf{v}_k) + \varepsilon}}\\ =& -\eta\mathbf{m}_k\left[\frac{1}{\sqrt{\mathbf{v}_k+\varepsilon}} + \sum_{i=1}^{+\infty}\frac{(-1)^i\binom{2i}{i}(1-\alpha)^i}{4^i(\mathbf{v}_k+\varepsilon)^{i+1/2}}(\mathbf{g}_k^2 - \mathbf{v}_k)^i\right]\\ =& -\eta\mathbf{m}_k\left[\frac{\alpha}{\sqrt{\mathbf{v}_k+\varepsilon}} + \frac{1-\alpha}{\sqrt{\mathbf{v}_k+\varepsilon}}\left(1 + \sum_{i=1}^{+\infty}\frac{(-1)^i\binom{2i}{i}(1-\alpha)^{i-1}}{4^i(\mathbf{v}_k+\varepsilon)^i}(\mathbf{g}_k^2 - \mathbf{v}_k)^i\right)\right]\\ =& \alpha\Delta_{2,k} - \eta\mathbf{m}_k\left(\frac{1}{\sqrt{\alpha\mathbf{v}_k + (1-\alpha)\mathbf{g}_k^2 + \varepsilon}} - \frac{\alpha}{\sqrt{\mathbf{v}_k+\varepsilon}}\right).\end{aligned}$$

thus $\mu_{2,k}=\alpha$, $\boldsymbol{f}_{2,k}(\mathbf{g}_k)=\mathbf{m}_k\left(\frac{1}{\sqrt{\alpha\mathbf{v}_k+(1-\alpha)\mathbf{g}_k^2+\varepsilon}}-\frac{\alpha}{\sqrt{\mathbf{v}_k+\varepsilon}}\right)$. According to Lemma 2, we obtain the corresponding $\boldsymbol{\Gamma}_k$ as

$$\boldsymbol{\Gamma}_{1,k}=\mu_{1,k}\boldsymbol{\Gamma}_{1,k-1}+\boldsymbol{f}_{1,k}(\mathbf{g})\approx\frac{\mathbf{g}}{\sqrt{\mathbf{v}_{k+1}+\varepsilon}} \tag{82}$$

and

$$\boldsymbol{\Gamma}_{2,k}=\mu_{2,k}\boldsymbol{\Gamma}_{2,k-1}+\boldsymbol{f}_{2,k}(\mathbf{g})\approx\frac{\mathbf{m}_k}{1-\alpha}\left(\frac{1}{\sqrt{\alpha\mathbf{v}_k+(1-\alpha)\mathbf{g}^2+\varepsilon}}-\frac{\alpha}{\sqrt{\mathbf{v}_k+\varepsilon}}\right). \tag{83}$$

So we can follow the calculation of $\boldsymbol{\gamma}_k$ and get $\boldsymbol{\gamma}_{1,k}$, $\boldsymbol{\gamma}_{2,k}$ based on $\boldsymbol{\Gamma}_{1,k}$, $\boldsymbol{\Gamma}_{2,k}$ respectively, then combine them to get the full $\boldsymbol{\gamma}_k$.

A.4.1 $\rho=2$.

When $\rho=2$, only $\boldsymbol{\gamma}_k^{(1)}$ needs to be calculated here. The only item in $\boldsymbol{\chi}_k^{(1)}$ is $q=2$, so the inner summation only needs to be performed on the set $\mathcal{S}_0^2=\{(\sigma_1=0,\sigma_2=0)\}$.

Since all operations in Adam are element-wise computations, we will use the component form for derivation in the following. Recall that x^i is the i-th component of vector $\boldsymbol{x}$, and the components of the ∇ operator are then denoted as $\nabla_i=\frac{\mathrm{d}}{\mathrm{d}\theta^i}$.

We first derive $\boldsymbol{\chi}_{1,j}^{(1)}$, there is

$$\chi_{1,j}^{(1),i}=\frac{1}{2}\left[\mathbf{L}_{1,j}^{(0)}\gamma_{1,j}^{(0),i}+\mu_{1,j}\mathbf{L}_{1,j-1}^{(0)}\gamma_{1,j-1}^{(0),i}\right]=\sum_u\frac{1}{2}\left[\Gamma_{1,j}^u\nabla_u\Gamma_{1,j}^i+\mu_{1,j}\Gamma_{1,j-1}^u\nabla_u\Gamma_{1,j-1}^i\right],$$

while $\nabla_u\Gamma_{1,j}^i=\frac{\nabla_u g^i}{\sqrt{v_j^i+\varepsilon}}$, thus

$$\chi_{1,j}^{(1),i}=\sum_u\frac{1}{2}\left[\frac{g^u}{\sqrt{v_j^u+\varepsilon}}\frac{\nabla_u g^i}{\sqrt{v_j^i+\varepsilon}}+\beta\frac{g^u}{\sqrt{v_{j-1}^u+\varepsilon}}\frac{\nabla_u g^i}{\sqrt{v_{j-1}^i+\varepsilon}}\right]. \tag{84}$$

When $k\to+\infty$, we take approximations $\mathbf{v}_k\to\mathbf{g}^2$ and $\mathbf{m}_k\to\mathbf{g}$, then

$$\chi_{1,j}^{(1),i}\approx\frac{1+\beta}{2\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{g^u\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}.$$

Whereupon we get

$$\gamma_{1,k}^{(1),i}=\sum_{j=0}^{k}\left(\prod_{p=j+1}^{k}\mu_{1,p}\right)\chi_{1,j}^{(1),i}\approx\frac{1+\beta}{2(1-\beta)\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{g^u\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}. \tag{85}$$

The derivation of $\boldsymbol{\gamma}_{2,k}^{(1)}$ proceeds in the same way. Firstly we have

$$\chi_{2,j}^{(1),i}=\frac{1}{2}\left[\mathbf{L}_{2,j}^{(0)}\gamma_{2,j}^{(0),i}+\mu_{2,j}\mathbf{L}_{2,j-1}^{(0)}\gamma_{2,j-1}^{(0),i}\right]=\sum_u\frac{1}{2}\left[\Gamma_{2,j}^u\nabla_u\Gamma_{2,j}^i+\mu_{2,j}\Gamma_{2,j-1}^u\nabla_u\Gamma_{2,j-1}^i\right].$$

Calculate $\nabla_u\Gamma_{2,j}^i$:

$$\nabla_u\Gamma_{2,j}^i=\frac{m_k^i}{1-\alpha}\nabla_u\left(\frac{1}{\sqrt{\alpha\mathbf{v}_j^i+(1-\alpha)(g^i)^2+\varepsilon}}-\frac{\alpha}{\sqrt{\mathbf{v}_j^i+\varepsilon}}\right)=-\frac{m_k^i g^i\nabla_u g^i}{(\alpha\mathbf{v}_j^i+(1-\alpha)(g^i)^2+\varepsilon)^{3/2}}.$$

Therefore

$$\Gamma_{2,j}^u\nabla_u\Gamma_{2,j}^i=-\frac{m_k^u m_k^i g^i\nabla_u g^i}{(1-\alpha)(\alpha v_j^i+(1-\alpha)(g^i)^2+\varepsilon)^{3/2}}\left(\frac{1}{\sqrt{\alpha v_j^u+(1-\alpha)(g^u)^2+\varepsilon}}-\frac{\alpha}{\sqrt{v_j^u+\varepsilon}}\right).$$

Substituting this into the expression for $\chi_{2,j}^{(1),i}$ and using the approximations $\lim_{k\to+\infty}\mathbf{v}_k=\mathbf{g}^2$ and $\lim_{k\to+\infty}\mathbf{m}_k=\mathbf{g}$ yields

$$\begin{aligned}\chi_{2,j}^{(1),i}=&\sum_u\frac{1}{2}\left[\Gamma_{2,j}^u\nabla_u\Gamma_{2,j}^i+\mu_{2,j}\Gamma_{2,j-1}^u\nabla_u\Gamma_{2,j-1}^i\right]\\=&-\sum_u\frac{1}{2}\left[\frac{m_k^u m_k^i g^i\nabla_u g^i}{(1-\alpha)(\alpha v_j^i+(1-\alpha)\left(g^i\right)^2+\varepsilon)^{3/2}}\left(\frac{1}{\sqrt{\alpha v_j^u+(1-\alpha)\left(g^u\right)^2+\varepsilon}}-\frac{\alpha}{\sqrt{v_j^u+\varepsilon}}\right)\right.\\&\left.+\frac{\alpha m_k^u m_k^i g^i\nabla_u g^i}{(1-\alpha)(\alpha v_{j-1}^i+(1-\alpha)\left(g^i\right)^2+\varepsilon)^{3/2}}\left(\frac{1}{\sqrt{\alpha v_{j-1}^u+(1-\alpha)\left(g^u\right)^2+\varepsilon}}-\frac{\alpha}{\sqrt{v_{j-1}^u+\varepsilon}}\right)\right]\\\approx&-\frac{1+\alpha}{2}\sum_u\left[\frac{g^u\left(g^i\right)^2\nabla_u g^i}{((g^i)^2+\varepsilon)^{3/2}\sqrt{(g^u)^2+\varepsilon}}\right]=-\frac{1+\alpha}{2}\frac{\left(g^i\right)^2}{((g^i)^2+\varepsilon)^{3/2}}\sum_u\frac{g^u\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}.\end{aligned}$$

Thus there is

$$\gamma_{2,k}^{(1),i}=\sum_{j=0}^{k}\chi_{2,j}^{(1),i}\prod_{p=j+1}^{k}\mu_{2,p}\approx-\frac{(1+\alpha)\left(g^i\right)^2}{2(1-\alpha)((g^i)^2+\varepsilon)^{3/2}}\sum_u\frac{g^u\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}.\tag{86}$$

Combining $\boldsymbol{\gamma}_{1,k}^{(1)}$ and $\boldsymbol{\gamma}_{2,k}^{(1)}$ gives the complete $\boldsymbol{\gamma}_k^{(1)}$

$$\begin{aligned}\gamma_k^{(1),i}=&\gamma_{1,k}^{(1),i}+\gamma_{2,k}^{(1),i}\\\approx&\frac{1+\beta}{2(1-\beta)\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{g^u\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}-\frac{(1+\alpha)\left(g^i\right)^2}{2(1-\alpha)((g^i)^2+\varepsilon)^{3/2}}\sum_u\frac{g^u\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}\\=&\frac{1}{2\sqrt{(g^i)^2+\varepsilon}}\left[\frac{1+\beta}{1-\beta}-\frac{(1+\alpha)\left(g^i\right)^2}{(1-\alpha)\left((g^i)^2+\varepsilon\right)}\right]\sum_u\frac{g^u\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}},\end{aligned}\tag{87}$$

which is consistent with the results of Cattaneo et al. (2024).

A.4.2 $\rho=3$.

When $\rho=3$, we need to continue calculating $\boldsymbol{\gamma}_k^{(2)}$ on the basis of the result of $\rho=2$. Now there are two terms in $\boldsymbol{\chi}_k^{(2)}$: $q=2$ and $q=3$, and the inner summation needs to be performed on the two sets respectively

$$\begin{aligned}\mathcal{S}_1^2=&\{(\sigma_1=1,\sigma_2=0),(\sigma_1=0,\sigma_2=1)\},\\\mathcal{S}_0^3=&\{(\sigma_1=0,\sigma_2=0,\sigma_3=0)\}.\end{aligned}$$

Hence the expression for $\boldsymbol{\chi}_j^{(2)}$ is now

$$\begin{aligned}\chi_j^{(2),i}=&\frac{1}{2}\left[\mathbf{L}_j^{(1)}\gamma_j^{(0),i}+\mu_j\mathbf{L}_{j-1}^{(1)}\gamma_{j-1}^{(0),i}\right]+\frac{1}{2}\left[\mathbf{L}_j^{(0)}\gamma_j^{(1),i}+\mu_j\mathbf{L}_{j-1}^{(0)}\gamma_{j-1}^{(1),i}\right]\\&+\frac{1}{6}\left[-\mathbf{L}_j^{(0)}\mathbf{L}_j^{(0)}\gamma_j^{(0),i}+\mu_j\mathbf{L}_{j-1}^{(0)}\mathbf{L}_{j-1}^{(0)}\gamma_{j-1}^{(0),i}\right]\\=&\frac{1}{2}\left[\sum_u\gamma_j^{(1),u}\nabla_u G_j^i+\mu_j\sum_u\gamma_{j-1}^{(1),u}\nabla_u\Gamma_{j-1}^i\right]+\frac{1}{2}\left[\sum_u G_j^u\nabla_u\gamma_j^{(1),i}+\mu_j\sum_u\Gamma_{j-1}^u\nabla_u\gamma_{j-1}^{(1),i}\right]\\&+\frac{1}{6}\left[-\sum_u G_j^u\nabla_u\left(\sum_v G_j^v\nabla_v G_j^i\right)+\mu_j\sum_u\Gamma_{j-1}^u\nabla_u\left(\sum_v\Gamma_{j-1}^v\nabla_v\Gamma_{j-1}^i\right)\right].\end{aligned}\tag{88}$$

The core of the derivation process that follows can be divided into three separate parts:

$$D_1=\sum_u\gamma_j^{(1),u}\nabla_u G_j^i,\ D_2=\sum_u G_j^u\nabla_u\gamma_j^{(1),i},\ D_3=\sum_u G_j^u\nabla_u\left(\sum_v G_j^v\nabla_v G_j^i\right).$$

Along the same process as when $\rho = 2$, we first settle $\boldsymbol{\chi}_{1,j}^{(2)}$. Using the results in $\rho = 2$ it is straightforward to obtain

$$\begin{aligned} D_{1,1} =& \sum_u \gamma_{1,j}^{(1),u} \nabla_u \Gamma_{1,j}^i \approx \sum_u \frac{1+\beta}{2(1-\beta)\sqrt{(g^u)^2+\varepsilon}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}} \frac{\nabla_u g^i}{\sqrt{(g^i)^2+\varepsilon}} \\ =& \frac{1+\beta}{2(1-\beta)\sqrt{(g^i)^2+\varepsilon}} \sum_u \frac{\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}}. \end{aligned}$$

The derivation of $D_{1,2}$ requires first computing $\nabla_u \boldsymbol{\gamma}_{1,j}^{(1)}$. Since

$$\gamma_{1,k}^{(1),i} = \sum_{j=0}^{k} \left(\prod_{p=j+1}^{k} \mu_{1,p} \right) \chi_{1,j}^{(1),i},$$

we have

$$\nabla_u \gamma_{1,k}^{(1),i} = \sum_{j=0}^{k} \beta^{k-j} \nabla_u \chi_{1,j}^{(1),i},$$

while

$$\begin{aligned} \nabla_u \chi_{1,j}^{(1),i} =& \nabla_u \left[\sum_v \frac{1}{2} \left(\Gamma_{1,j}^v \nabla_v \Gamma_{1,j}^i + \mu_{1,j} \Gamma_{1,j-1}^v \nabla_v \Gamma_{1,j-1}^i \right) \right] \\ =& \frac{1}{2} \left[\nabla_u \left(\sum_v \Gamma_{1,j}^v \nabla_v \Gamma_{1,j}^i \right) + \mu_{1,j} \nabla_u \left(\sum_v \Gamma_{1,j-1}^v \nabla_v \Gamma_{1,j-1}^i \right) \right]. \end{aligned}$$

Denote that $W_{1,j}^{(u,v,i)} = \nabla_u \left(\Gamma_{1,j}^v \nabla_v \Gamma_{1,j}^i \right)$. This term also occurs in $D_{1,3}$, therefore we first need to find $W_{1,j}^{(u,v,i)}$.

Expanding $W_{1,j}^{(u,v,i)} = \nabla_u \Gamma_{1,j}^v \nabla_v \Gamma_{1,j}^i + \Gamma_{1,j}^v \nabla_u \nabla_v \Gamma_{1,j}^i$, where the first term can be obtained directly using the existing results

$$\nabla_u \Gamma_{1,j}^v \nabla_v \Gamma_{1,j}^i = \frac{\nabla_u g^v}{\sqrt{v_j^v + \varepsilon}} \frac{\nabla_v g^i}{\sqrt{v_j^i + \varepsilon}} \approx \frac{\nabla_u g^v \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}\sqrt{(g^i)^2+\varepsilon}}.$$

In the second term

$$\nabla_u \nabla_v \Gamma_{1,j}^i = \frac{\nabla_u \nabla_v g^i}{\sqrt{v_j^i + \varepsilon}} \approx \frac{\nabla_u \nabla_v g^i}{\sqrt{(g^i)^2+\varepsilon}},$$

so that

$$\Gamma_{1,j}^v \nabla_u \nabla_v \Gamma_{1,j}^i \approx \frac{g^v \nabla_u \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}\sqrt{(g^i)^2+\varepsilon}}.$$

Therefore, we get

$$W_{1,j}^{(u,v,i)} \approx \frac{\nabla_u g^v \nabla_v g^i + g^v \nabla_u \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}\sqrt{(g^i)^2+\varepsilon}}.$$

Applying $W_{1,j}^{(u,v,i)}$ step by step to $\nabla_u \boldsymbol{\chi}_{1,j}^{(1)}$, $\nabla_u \boldsymbol{\gamma}_{1,j}^{(1)}$, $D_{1,2}$ and $D_{1,3}$, we first get

$$\nabla_u \chi_{1,j}^{(1),i} = \frac{1}{2} \left[\sum_v W_{1,j}^{(u,v,i)} + \mu_{1,j} \sum_v W_{1,j-1}^{(u,v,i)} \right] \approx \frac{1+\beta}{2} \sum_v \frac{\nabla_u g^v \nabla_v g^i + g^v \nabla_u \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}\sqrt{(g^i)^2+\varepsilon}}$$

$$=\frac{1+\beta}{2\sqrt{(g^i)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}}.$$

Then

$$\nabla_u\gamma_{1,k}^{(1),i}=\sum_{j=0}^{k}\beta^{k-j}\nabla_u\chi_{1,j}^{(1),i}\approx\frac{1+\beta}{2(1-\beta)\sqrt{(g^i)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}}.$$

Therefore we have

$$D_{1,2}=\sum_u\Gamma_{1,j}^u\nabla_u\gamma_{1,j}^{(1),i}\approx\frac{1+\beta}{2(1-\beta)\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{g^u}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}},$$

and

$$D_{1,3}=\sum_u\Gamma_{1,j}^u\sum_v W_{1,j}^{(u,v,i)}\approx\frac{1}{\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{g^u}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}}.$$

Applying all the above results to $\chi_{1,j}^{(2),i}$, we have

$$\begin{aligned}
\chi_{1,j}^{(2),i}\approx&\frac{1+\beta}{2}D_{1,1}+\frac{1+\beta}{2}D_{1,2}-\frac{1-\beta}{6}D_{1,3}\\
\approx&\frac{1+\beta}{2}\frac{1+\beta}{2(1-\beta)\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{g^v\nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}}\\
&+\frac{1+\beta}{2}\frac{1+\beta}{2(1-\beta)\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{g^u}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}}\\
&-\frac{1-\beta}{6}\frac{1}{\sqrt{(g^i)^2+\varepsilon}}\sum_u\frac{g^u}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}}\\
=&\frac{(1+\beta)^2}{4(1-\beta)\sqrt{(g^i)^2+\varepsilon}}\left[\sum_u\frac{\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{g^v\nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}}\right.\\
&\left.+\frac{1+10\beta+\beta^2}{3(1+\beta)^2}\sum_u\frac{g^u}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}}\right].
\end{aligned}$$

Thus

$$\begin{aligned}
\gamma_{1,k}^{(2),i}=&\sum_{j=0}^{k}\left(\prod_{p=j+1}^{k}\mu_{1,p}\right)\chi_{1,j}^{(2),i}\\
\approx&\frac{(1+\beta)^2}{4(1-\beta)^2\sqrt{(g^i)^2+\varepsilon}}\left[\sum_u\frac{\nabla_u g^i}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{g^v\nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}}\right.\\
&\left.+\frac{1+10\beta+\beta^2}{3(1+\beta)^2}\sum_u\frac{g^u}{\sqrt{(g^u)^2+\varepsilon}}\sum_v\frac{\nabla_u g^v\nabla_v g^i+g^v\nabla_u\nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}}\right]
\end{aligned}\tag{89}$$

Proceeding in the steps above similarly leads to $\gamma_{2,k}^{(2),i}$. The procedure is not repeated here, but only the key results are given in the order in which they are calculated. The definitions of the variables are the same as above.

$D_{2,1}$ can also be derived by results in $\rho = 2$, thus we get

$$D_{2,1} = \sum_u \gamma_{2,j}^{(1),u} \nabla_u \Gamma_{2,j}^i \approx \sum_u \frac{(1+\alpha)\left(g^u\right)^2}{2(1-\alpha)((g^u)^2+\varepsilon)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}} \frac{\left(g^i\right)^2 \nabla_u g^i}{((g^i)^2+\varepsilon)^{3/2}}$$
$$= \frac{(1+\alpha)\left(g^i\right)^2}{2(1-\alpha)((g^i)^2+\varepsilon)^{3/2}} \sum_u \frac{\left(g^u\right)^2 \nabla_u g^i}{((g^u)^2+\varepsilon)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}}.$$

To get $W_{2,j}^{(u,v,i)}$, we first deal with $\nabla_u \Gamma_{2,j}^v \nabla_v \Gamma_{2,j}^i$

$$\nabla_u \Gamma_{2,j}^v \nabla_v \Gamma_{2,j}^i \approx \frac{\left(g^v\right)^2 \nabla_u g^v \left(g^i\right)^2 \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}((g^i)^2+\varepsilon)^{3/2}},$$

then we get $\nabla_u \nabla_v \Gamma_{2,j}^i$ based on the results of $\nabla_v \Gamma_{2,j}^i$, which is

$$\nabla_u \nabla_v \Gamma_{2,j}^i = -\nabla_u \frac{m_k^i g^i \nabla_v g^i}{(\alpha v_j^i + (1-\alpha)\left(g^i\right)^2 + \varepsilon)^{3/2}}$$
$$= \frac{3m_k^i (g^i \nabla_v g^i) \nabla_u (\alpha v_j^i + (1-\alpha)\left(g^i\right)^2 + \varepsilon)}{2(\alpha v_j^i + (1-\alpha)\left(g^i\right)^2 + \varepsilon)^{5/2}} - \frac{m_k^i \nabla_u (g^i \nabla_v g^i)}{(\alpha v_j^i + (1-\alpha)\left(g^i\right)^2 + \varepsilon)^{3/2}}$$
$$\approx \frac{3(1-\alpha) g^{(i)3} \nabla_v g^i \nabla_u g^i}{((g^i)^2+\varepsilon)^{5/2}} - \frac{g^i \nabla_u (g^i \nabla_v g^i)}{((g^i)^2+\varepsilon)^{3/2}},$$

hence

$$\Gamma_{2,j}^v \nabla_u \nabla_v \Gamma_{2,j}^i \approx \frac{3(1-\alpha) g^v \left(g^{(i)}\right)^3 \nabla_v g^i \nabla_u g^i}{\sqrt{(g^v)^2+\varepsilon}((g^i)^2+\varepsilon)^{5/2}} - \frac{g^v g^i \nabla_u (g^i \nabla_v g^i)}{\sqrt{(g^v)^2+\varepsilon}((g^i)^2+\varepsilon)^{3/2}}.$$

Thus there is

$$W_{2,j}^{(u,v,i)} \approx \frac{\left(g^v\right)^2 \nabla_u g^v \left(g^i\right)^2 \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}((g^i)^2+\varepsilon)^{3/2}} + \frac{3(1-\alpha) g^v \left(g^{(i)}\right)^3 \nabla_v g^i \nabla_u g^i}{\sqrt{(g^v)^2+\varepsilon}((g^i)^2+\varepsilon)^{5/2}} - \frac{g^v g^i \nabla_u (g^i \nabla_v g^i)}{\sqrt{(g^v)^2+\varepsilon}((g^i)^2+\varepsilon)^{3/2}}.$$

Substituting $W_{2,j}^{(u,v,i)}$ into $\nabla_u \chi_{2,j}^{(1),i}$ yields

$$\nabla_u \chi_{2,j}^{(1),i} = \frac{1}{2}\left[\sum_v W_{2,j}^{(u,v,i)} + \mu_{2,j} \sum_v W_{2,j-1}^{(u,v,i)}\right] \approx \frac{1+\alpha}{2} \sum_v W_{2,j}^{(u,v,i)}$$
$$\approx \frac{(1+\alpha)\left(g^i\right)^2}{2((g^i)^2+\varepsilon)^{3/2}} \left[\sum_v \frac{\left(g^v\right)^2 \nabla_u g^v \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}} + \frac{3(1-\alpha) g^i \nabla_u g^i}{(g^i)^2+\varepsilon} \sum_v \frac{g^v \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}} - \sum_v \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{(g^v)^2+\varepsilon}}\right].$$

Thus

$$\nabla_u \gamma_{2,k}^{(1),i} \approx \frac{1+\alpha}{2(1-\alpha)} \sum_v W_{2,j}^{(u,v,i)}$$
$$\approx \frac{(1+\alpha)\left(g^i\right)^2}{2(1-\alpha)((g^i)^2+\varepsilon)^{3/2}} \left[\sum_v \frac{\left(g^v\right)^2 \nabla_u g^v \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}} + \frac{3(1-\alpha) g^i \nabla_u g^i}{(g^i)^2+\varepsilon} \sum_v \frac{g^v \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}} - \sum_v \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{(g^v)^2+\varepsilon}}\right].$$

As a result, we get

$$D_{2,2} = \sum_u \Gamma_{2,j}^u \nabla_u \gamma_{2,j}^{(1),i} \approx \frac{1+\alpha}{2(1-\alpha)} \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \sum_v W_{2,j}^{(u,v,i)}$$

$$\approx \frac{(1+\alpha)\left(g^i\right)^2}{2(1-\alpha)((g^i)^2+\varepsilon)^{3/2}} \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \left[\sum_v \frac{(g^v)^2 \nabla_u g^v \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}} + \frac{3(1-\alpha) g^i \nabla_u g^i}{(g^i)^2+\varepsilon} \sum_v \frac{g^v \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}} - \sum_v \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{(g^v)^2+\varepsilon}} \right],$$

and

$$\begin{aligned} D_{2,3} =& \sum_u \Gamma^u_{2,j} \sum_v W^{(u,v,i)}_{2,j} \approx \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \sum_v W^{(u,v,i)}_{2,j} \\ \approx& \frac{\left(g^i\right)^2}{((g^i)^2+\varepsilon)^{3/2}} \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \left[\sum_v \frac{(g^v)^2 \nabla_u g^v \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}} + \frac{3(1-\alpha) g^i \nabla_u g^i}{(g^i)^2+\varepsilon} \sum_v \frac{g^v \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}} - \sum_v \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{(g^v)^2+\varepsilon}} \right]. \end{aligned}$$

Therefore

$$\begin{aligned} \chi^{(2),i}_{2,j} \approx& \frac{1+\alpha}{2} D_{2,1} + \frac{1+\alpha}{2} D_{2,2} - \frac{1-\alpha}{6} D_{2,3} \\ \approx& \frac{(1+\alpha)^2\left(g^i\right)^2}{4(1-\alpha)((g^i)^2+\varepsilon)^{3/2}} \sum_u \frac{(g^u)^2 \nabla_u g^i}{((g^u)^2+\varepsilon)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}} + \frac{1+10\alpha+\alpha^2}{12(1-\alpha)} \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \sum_v W^{(u,v,i)}_{2,j} \\ \approx& \frac{(1+\alpha)^2\left(g^i\right)^2}{4(1-\alpha)((g^i)^2+\varepsilon)^{3/2}} \left\{ \sum_u \frac{(g^u)^2 \nabla_u g^i}{((g^u)^2+\varepsilon)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}} \right. \\ & + \frac{1+10\alpha+\alpha^2}{3(1+\alpha)^2} \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \left[\sum_v \frac{(g^v)^2 \nabla_u g^v \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}} \right. \\ & \left. \left. + \frac{3(1-\alpha) g^i \nabla_u g^i}{(g^i)^2+\varepsilon} \sum_v \frac{g^v \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}} - \sum_v \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{(g^v)^2+\varepsilon}} \right] \right\}, \end{aligned}$$

and

$$\begin{aligned} \gamma^{(2),i}_{2,k} \approx& \frac{(1+\alpha)^2\left(g^i\right)^2}{4(1-\alpha)^2((g^i)^2+\varepsilon)^{3/2}} \sum_u \frac{(g^u)^2 \nabla_u g^i}{((g^u)^2+\varepsilon)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}} + \frac{1+10\alpha+\alpha^2}{12(1-\alpha)^2} \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \sum_v W^{(u,v,i)}_{2,j} \\ \approx& \frac{(1+\alpha)^2\left(g^i\right)^2}{4(1-\alpha)^2((g^i)^2+\varepsilon)^{3/2}} \left\{ \sum_u \frac{(g^u)^2 \nabla_u g^i}{((g^u)^2+\varepsilon)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{(g^v)^2+\varepsilon}} \right. \\ & + \frac{1+10\alpha+\alpha^2}{3(1+\alpha)^2} \sum_u \frac{g^u}{\sqrt{(g^u)^2+\varepsilon}} \left[\sum_v \frac{(g^v)^2 \nabla_u g^v \nabla_v g^i}{((g^v)^2+\varepsilon)^{3/2}} \right. \\ & \left. \left. + \frac{3(1-\alpha) g^i \nabla_u g^i}{(g^i)^2+\varepsilon} \sum_v \frac{g^v \nabla_v g^i}{\sqrt{(g^v)^2+\varepsilon}} - \sum_v \frac{g^v \nabla_u (g^i \nabla_v g^i)}{g^i \sqrt{(g^v)^2+\varepsilon}} \right] \right\}. \end{aligned} \tag{90}$$

Finally, we combine $\gamma^{(2),i}_{1,k}$ and $\gamma^{(2),i}_{2,k}$ to obtain an expression for $\gamma^{(2),i}_k$

$$\gamma^{(2),i}_k = \gamma^{(2),i}_{1,k} + \gamma^{(2),i}_{2,k}$$

$$
\begin{aligned}
\approx & \frac{(1+\beta)^2}{4(1-\beta)^2\sqrt{\left(g^i\right)^2+\varepsilon}}\left[\sum_u \frac{\nabla_u g^i}{\sqrt{\left(g^u\right)^2+\varepsilon}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{\left(g^v\right)^2+\varepsilon}}\right. \\
& \left.+\frac{1+10\beta+\beta^2}{3(1+\beta)^2} \sum_u \frac{g^u}{\sqrt{\left(g^u\right)^2+\varepsilon}} \sum_v \frac{\nabla_u g^v \nabla_v g^i+g^v \nabla_u \nabla_v g^i}{\sqrt{\left(g^v\right)^2+\varepsilon}}\right] \\
& +\frac{(1+\alpha)^2\left(g^i\right)^2}{4(1-\alpha)^2\left(\left(g^i\right)^2+\varepsilon\right)^{3/2}}\left\{\sum_u \frac{\left(g^u\right)^2 \nabla_u g^i}{\left(\left(g^u\right)^2+\varepsilon\right)^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{\left(g^v\right)^2+\varepsilon}}\right. \\
& +\frac{1+10\alpha+\alpha^2}{3(1+\alpha)^2} \sum_u \frac{g^u}{\sqrt{\left(g^u\right)^2+\varepsilon}}\left[\sum_v \frac{\left(g^v\right)^2 \nabla_u g^v \nabla_v g^i}{\left(\left(g^v\right)^2+\varepsilon\right)^{3/2}}\right. \\
& \left.\left.\left.+\frac{3(1-\alpha) g^i \nabla_u g^i}{\left(g^i\right)^2+\varepsilon} \sum_v \frac{g^v \nabla_v g^i}{\sqrt{\left(g^v\right)^2+\varepsilon}}-\sum_v \frac{g^v \nabla_u\left(g^i \nabla_v g^i\right)}{g^i \sqrt{\left(g^v\right)^2+\varepsilon}}\right]\right\} .
\end{aligned}
\tag{91}
$$

A.4.3 Relationship between Adam and GD/HB

Here a further corollary is given which indicates the relationship between Adam and HB/GD.

Corollary 1.5 (Degradation of Adam in the case of large ε)**.** *Assume that the loss function is L, the learning rate is η and the hyperparameters of Adam is α, β, which are all defined as in Eq.* (11)*. If ε takes on a value such that $\varepsilon \gg \left(g^i\right)^2$ always hold for all i, then when k is large enough, Adam degenerates into HB with the loss function $L' = (1-\beta)L$, learning rate $\eta' = \frac{\eta}{\sqrt{\varepsilon}}$ and the hyperparameter $\beta' = \beta$.*

Proof. When $\left(g^i\right)^2 \ll \varepsilon$, the iterative equation of Adam could be simplified as

$$
\boldsymbol{\theta}_{k+1}-\boldsymbol{\theta}_k=-\eta \frac{\hat{\mathbf{m}}_{k+1}}{\sqrt{\hat{\mathbf{v}}_{k+1}+\varepsilon}} \approx-\frac{\eta}{\sqrt{\varepsilon}}\left(\beta \mathbf{m}_k+(1-\beta) \mathbf{g}_k\right).
$$

We take

$$
\eta'=\frac{\eta}{\sqrt{\varepsilon}}, \quad L'=(1-\beta) L, \quad \mathbf{g}_k'=\nabla L'=(1-\beta) \mathbf{g}_k,
$$

then $\boldsymbol{\theta}_{k+1}-\boldsymbol{\theta}_k \approx-\eta'\left(\beta \mathbf{m}_k+\mathbf{g}_k'\right)=-\eta' \mathbf{g}_k'+\beta\left(\boldsymbol{\theta}_k-\boldsymbol{\theta}_{k-1}\right)$. This is actually an optimization using the HB method on the loss function $(1-\beta)L$, with the hyper parameters taken as the learning rate $\eta/\sqrt{\varepsilon}$ and the hyperparameter β. □

With Corollary 1.5, we can conveniently build continuous approximations for HB and GD back after obtaining the continuous approximation for Adam.

For example, if $\left(g^i\right)^2 \ll \varepsilon$, then Eq. (87) could be simplified as

$$
\gamma_k^{(1), i} \approx \frac{1}{2\sqrt{\varepsilon}}\left[\frac{1+\beta}{1-\beta}-\frac{(1+\alpha)\left(g^i\right)^2}{(1-\alpha)\varepsilon}\right] \sum_u \frac{g^u \nabla_u g^i}{\sqrt{\varepsilon}} \approx \frac{1+\beta}{2(1-\beta)\varepsilon} \sum_u g^u \nabla_u g^i, \tag{92}
$$

and Eq. (91) could be simplified as

$$
\begin{aligned}
\gamma_k^{(2), i} \approx & \frac{(1+\beta)^2}{4(1-\beta)^2\sqrt{\varepsilon}}\left[\sum_u \frac{\nabla_u g^i}{\sqrt{\varepsilon}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{\varepsilon}}+\frac{1+10\beta+\beta^2}{3(1+\beta)^2} \sum_u \frac{g^u}{\sqrt{\varepsilon}} \sum_v \frac{\nabla_u g^v \nabla_v g^i+g^v \nabla_u \nabla_v g^i}{\sqrt{\varepsilon}}\right] \\
& +\frac{(1+\alpha)^2\left(g^i\right)^2}{4(1-\alpha)^2 \varepsilon^{3/2}}\left\{\sum_u \frac{\left(g^u\right)^2 \nabla_u g^i}{\varepsilon^{3/2}} \sum_v \frac{g^v \nabla_v g^u}{\sqrt{\varepsilon}}\right.
\end{aligned}
$$

$$+\frac{1+10\alpha+\alpha^2}{3(1+\alpha)^2}\sum_u \frac{g^u}{\sqrt{\varepsilon}}\left[\sum_v \frac{\left(g^v\right)^2\nabla_u g^v\nabla_v g^i}{\varepsilon^{3/2}}-\frac{\nabla_u g^i}{g^i}\sum_v\frac{g^v\nabla_v g^i}{\sqrt{\varepsilon}}-\sum_v\frac{g^v\nabla_u\nabla_v g^i}{\sqrt{\varepsilon}}\right]\Bigg\}$$

$$\approx \frac{(1+\beta)^2}{4(1-\beta)^2\varepsilon^{3/2}}\left[\sum_u\sum_v \nabla_u g^i g^v\nabla_v g^u+\frac{1+10\beta+\beta^2}{3(1+\beta)^2}\sum_u\sum_v g^u\nabla_u g^v\nabla_v g^i\right]. \tag{93}$$

Comparing Eq. (92) with Eq. (71), Eq. (93) with Eq. (79), it can be observed that they use the gradient for the calculation in exactly the same way, and the differences in the coefficients are fully consistent with Corollary 1.5. Therefore, we actually only need to utilize our framework to analyze Adam, and the conclusions of the simpler methods, such as GD or HB, can be easily restored. From Eq. (92) and Eq. (93), the dynamics at the end of optimization of Adam depend almost entirely on the values taken for β and ε.

B Implicit Bias of Gradient-based Methods embodied in higher-order continuous approximations

To clarify the dynamics that this approximation term represents in the learning process, we take some reasonable approximations to further simplify its form.

B.1 Implicit bias of Adam if $\left(g^i\right)^2 \gg \varepsilon$

In this section, we use our continuous approximation for Adam with discretization error to $\mathcal{O}(\eta^3)$ to analyze its additional implicit bias. For lower order cases, since the continuous approximation has the same form as that of Cattaneo et al. (2024), the conclusions are the same and will not be developed here.

When $\left(g^i\right)^2 \gg \varepsilon$, we can ignore the ε in most places and there is a approximate relationship

$$\sqrt{\left(g^i\right)^2+\varepsilon}\approx |g^i|. \tag{94}$$

The numerical error of the approximation of Eq. (94) for $\left(g^i\right)^2 \gg \varepsilon$ is very small, so we can use this approximation forward to greatly simplify the arithmetic. Besides, as $\left(g^i\right)^2 \approx 0$, we will also use this approximation backward in order to avoid a denominator of 0 or a logarithmic function acting on 0. With this approximation, we can also denote another symbol as

$$\operatorname{sgn}\left(g^i\right)=\frac{g^i}{|g^i|}\approx\frac{g^i}{\sqrt{\left(g^i\right)^2+\varepsilon}}. \tag{95}$$

Usually in practice, ε is taken on a very small value(for example, $\varepsilon = 10^{-8}$), with the greatest significance being to prevent overflow due to division by zero. Moreover, during the training of large-scale neural networks, we generally do not seek to converge to a strict minima, so for the optimizers, the gradient is usually not zero even to the end of training, thus $\left(g^i\right)^2 \gg \varepsilon$ holds most of the time. This situation is the main factor affecting the implicit bias of Adam.

In most practical cases, we take the parameter $\alpha \approx 1$, which means $1-\alpha \approx 0$, therefore we could ignore the term with $1-\alpha$ on the numerator. Thus from Eq. (91) there is

$$\begin{aligned}\gamma_k^{(2),i}\approx&\frac{(1+\beta)^2}{4(1-\beta)^2|g^i|}\sum_u\sum_v\left[\frac{\nabla_u g^i}{|g^u|}\operatorname{sgn}\left(g^v\right)\nabla_v g^u\right.\\&\left.+\frac{1+10\beta+\beta^2}{3(1+\beta)^2}\operatorname{sgn}\left(g^u\right)\left(\frac{\nabla_u g^v\nabla_v g^i}{|g^v|}+\operatorname{sgn}\left(g^v\right)\nabla_u\nabla_v g^i\right)\right]\\&+\frac{(1+\alpha)^2}{4(1-\alpha)^2|g^i|}\sum_u\sum_v\left[\frac{\nabla_u g^i}{|g^u|}\operatorname{sgn}\left(g^v\right)\nabla_v g^u\right.\\&\left.+\frac{1+10\alpha+\alpha^2}{3(1+\alpha)^2}\operatorname{sgn}\left(g^u\right)\left(\frac{\nabla_u g^v\nabla_v g^i}{|g^v|}-\operatorname{sgn}\left(g^v\right)\nabla_u\nabla_v g^i-\frac{\operatorname{sgn}\left(g^v\right)\nabla_u g^i\nabla_v g^i}{g^i}\right)\right].\end{aligned} \tag{96}$$

Obviously, there are 4 distinct components in Eq. (96):

$$\begin{cases} P_1 = \sum_u \sum_v \frac{\nabla_u g^i}{|g^u|} \operatorname{sgn}(g^v) \nabla_v g^u \\ P_2 = \sum_u \sum_v \operatorname{sgn}(g^u) \frac{\nabla_u g^v \nabla_v g^i}{|g^v|} \\ P_3 = \sum_u \sum_v \operatorname{sgn}(g^u) \operatorname{sgn}(g^v) \nabla_u \nabla_v g^i \\ P_4 = \sum_u \sum_v \frac{\operatorname{sgn}(g^v) \nabla_u g^i \nabla_v g^i}{g^i} \end{cases}.$$

It can be easily observed that $P_1 = P_2$, since the positions of u and v in these two equations are symmetrical, so that P_1 and P_2 are practically identical after summing u and v. Meanwhile, for P_4, since the sign of g^i can be regarded as completely random, the sign of P_4 is actually also unstable and cannot form a stable implicit bias. Therefore, the only thing we need to consider is the implicit bias represented by P_1 and P_3.

Since we assume that the loss function L and its derivatives of all orders are uniformly continuous, the order of differentiation is always exchangeable for L, thus for P_3, there is

$$P_3 = \nabla_i \sum_u \sum_v \operatorname{sgn}(g^u) \operatorname{sgn}(g^v) \nabla_u g^v = \nabla_i \left[\operatorname{sgn}(\mathbf{g}) \cdot \nabla \mathbf{g} \cdot \operatorname{sgn}(\mathbf{g}) \right].$$

Meanwhile in the $\gamma_k^{(2),i}$, the coefficients of P_3 is

$$\frac{(1+\beta)^2}{4(1-\beta)^2\sqrt{(g^i)^2+\varepsilon}}\zeta(\beta) - \frac{(1+\alpha)^2}{4(1-\alpha)^2\sqrt{(g^i)^2+\varepsilon}}\zeta(\alpha). \tag{97}$$

Ma et al. (2022) states that Adam with hyperparameter $\alpha < \beta$ is unstable, so in most practices there is always $\alpha > \beta$, thus Eq. (97) is negative. This means that the regularization represented by P_3 is

$$R_3 = -\operatorname{sgn}(\mathbf{g}) \cdot \nabla \mathbf{g} \cdot \operatorname{sgn}(\mathbf{g}),$$

i.e., it implicitly encourages $\operatorname{sgn}(\mathbf{g}) \cdot \nabla \mathbf{g} \cdot \operatorname{sgn}(\mathbf{g})$ to increase.

Inspired by the work of Rosca et al. (2023), we follow a similar idea of ignoring the effect of terms associated with second-order and higher derivatives of the gradient in the process, then we can assemble P_1 into the following total differential form

$$\begin{aligned} P_1 \approx & \nabla_i \sum_u \sum_v \left[\ln |g^u| \operatorname{sgn}(g^u) \nabla_u g^v \operatorname{sgn}(g^v) \right] \\ = & \nabla_i \left[(\ln |\mathbf{g}| \odot \operatorname{sgn}(\mathbf{g})) \cdot \nabla \mathbf{g} \cdot \operatorname{sgn}(\mathbf{g}) \right] \qquad (98) \\ \approx & \nabla_i \left[\left(\ln \sqrt{\mathbf{g}^2 + \varepsilon} \odot \operatorname{sgn}(\mathbf{g}) \right) \cdot \nabla \mathbf{g} \cdot \operatorname{sgn}(\mathbf{g}) \right]. \qquad (99) \end{aligned}$$

where we specifically use "$\odot$" to show that here is element-wise multiplication. Eq. (98) to Eq. (99) we use Eq. (94) backward to avoid the logarithmic function acting on 0. Meanwhile in the $\gamma_k^{(2),i}$, the coefficients of P_1 (and P_2 included together) is

$$\left[\frac{(1+\beta)^2}{4(1-\beta)^2\sqrt{(g^i)^2+\varepsilon}} + \frac{(1+\alpha)^2}{4(1-\alpha)^2\sqrt{(g^i)^2+\varepsilon}} \right] (1+\zeta(\alpha)). \tag{100}$$

Eq. (100) is always positive. Therefore, the practical effect of P_1 is to minimize

$$R_1 = \left(\ln \sqrt{\mathbf{g}^2 + \varepsilon} \odot \operatorname{sgn}(\mathbf{g}) \right) \cdot \nabla \mathbf{g} \cdot \operatorname{sgn}(\mathbf{g})$$

R_1 looks somewhat similar to R_3, but its form is not entirely the same. However, we can qualitatively state that both of them lead to similar regularization.

We focus on the main difference between R_1 and R_3, which is $\ln\sqrt{\mathbf{g}^2+\varepsilon}$. In fact, for common activation functions such as $ReLU$, $tanh$, $sigmoid$, their derivatives are always less than or equal to 1. For softmax, its derivative depends on the output of the layer, and most of the neural networks perform batch or layer normalization, so the absolute value of the derivative of softmax is also less than 1 most of the time. Even if the absolute value of some of the initial gradient component $\left(g^i\right)^2$ is greater than 1, it usually decreases to less than 1 as the optimization proceeds, since the goal of the optimizer is to find the optimal point where $\mathbf{g}=0$. Therefore, we may assume that during the optimization, there is always $\left(g^i\right)^2<1$, and thus $\ln\sqrt{\left(g^i\right)^2+\varepsilon}<0$. Moreover, $\ln\sqrt{\left(g^i\right)^2+\varepsilon}$ has a lower bound of $\frac{1}{2}\ln\varepsilon$ (e.g., when $\varepsilon=10^{-8}$, we have $\ln\sqrt{\left(g^i\right)^2+\varepsilon}\geq -9.21$ for the ε-inside form of Adam; and for the ε-outside form of Adam, there is $\ln\left(\sqrt{\left(g^i\right)^2}+\varepsilon\right)\geq -18.42$).

Therefore, we can regard $\ln\sqrt{\left(g^i\right)^2+\varepsilon}$ as some kind of weighting coefficient with a limited range of variation, which gives relatively more importance to the regularization effect brought about by g^i with smaller absolute values, while it basically does not bring additional regularization for the components which satisfy $\left(g^i\right)^2\approx 1$.

Thus in general, R_1 can be regarded as a weighted regularization of $\operatorname{sgn}(\mathbf{g})\cdot\nabla\mathbf{g}\cdot\operatorname{sgn}(\mathbf{g})$. Since $\ln\sqrt{\left(g^i\right)^2+\varepsilon}<0$, the minimization of R_1 will be close to the maximization of $\operatorname{sgn}(\mathbf{g})\cdot\nabla\mathbf{g}\cdot\operatorname{sgn}(\mathbf{g})$, so its actual regularization direction is similar to that of R_3. Here we can verify this by observing the ratio of R_1 to R_3 in the experiments of Section 3.2, as shown in Fig. 1. It can be clearly seen that the value of R_1/R_3 varies completely with the increase or decrease of the norm of the gradient (when the norm decrease, R_1/R_3 increase). Meanwhile, the value range of R_1/R_3 is also basically fixed and consistent with our estimation above (when ε increase, the maximum of R_1/R_3 decrease, and the maximum always comes when the norm of the gradient is small).

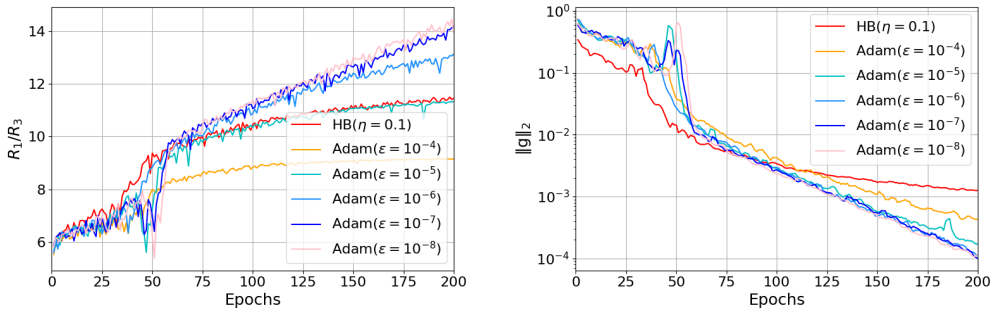


Figure 1: Variation of R_1/R_3 in the experiments of Fig. 5, Section 3.2.

In summary, we can consider that the implicit bias implied in the third order approximation of Adam when $\left(g^i\right)^2\gg\varepsilon$ is

$$R_a=-\operatorname{sgn}(\mathbf{g})\cdot\nabla\mathbf{g}\cdot\operatorname{sgn}(\mathbf{g}). \tag{101}$$

Based on our derivation process and corollary, we can try to give a regularization perspective to the conclusion of Ma et al. (2022), which is that when $\alpha<\beta$, the signs of R_1 and R_3 will no longer be the same, which will lead to unanticipated regularization, thus affecting the stability of convergence of Adam.

B.2 Implicit bias of Adam if $\left(\mathbf{g}^i\right)^2\ll\varepsilon$

In Appendix A.4.3, we have got the conclusion when Adam degenerates to non-adaptive learning rate methods by taking $\varepsilon\gg\left(g^i\right)^2$, and the form of continuous approximation is consistent with the

HB method. We take similar assumptions as in Appendix B.1, when $\beta \approx 1$, we have $\zeta(\beta) \approx 1$, so the regularization term could be written as

$$P_c = \sum_u \sum_v \nabla_u g^i g^v \nabla_v g^u + \sum_u \sum_v g^u \nabla_u g^v \nabla_v g^i \approx \nabla_i \left(\mathbf{g} \cdot \nabla \mathbf{g} \cdot \mathbf{g}\right).$$

And its coefficient $\frac{(1+\beta)^2}{4(1-\beta)^2\varepsilon^{3/2}}$ is always positive, thus the regularization effect is to minimize

$$R_g = \mathbf{g} \cdot \nabla \mathbf{g} \cdot \mathbf{g}. \tag{102}$$

Notice that the regularization of Adam at $\left(\mathbf{g}^i\right)^2 \ll \varepsilon$ is similar to that of HB, which confirms our Corollary 1.5, i.e., the degeneration of Adam to HB at $\left(\mathbf{g}^i\right)^2 \ll \varepsilon$.

C Details for Numerical Experiments

The experiments are conducted on a Windows 11 platform equipped with an AMD® Ryzen 7 7700 CPU at 3.00 GHz, 32GB of RAM, and an NVIDIA RTX 4070 graphics card.

C.1 Details for Section 3.1

For the discrete Adam optimizer, the step size is set as η. For the continuous approximations, $\eta_{\text{Euler}} = \eta/10$ is set as the Euler step sizes to approximate the dynamics. All the hyper-parameters are listed in Table 3.

x, y	1, 0.6
Starting point	$a_1 = 1, a_2 = 1.3$
η	1×10^{-5}
ε	1×10^{-3}
α	0.999
β	0.9
η_{Euler}	1×10^{-6}

Table 3: Hyper-parameters for 2-d model.

The model parameter is denoted as $\boldsymbol{\theta} = (\boldsymbol{\theta}^1, \boldsymbol{\theta}^2)^T \in \mathbb{R}^2$. For the continuous approximations, we use the ODE

$$\dot{\boldsymbol{\theta}} = F_\rho \implies \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \eta_{\text{Euler}} F_\rho.$$

Experiment: Discretization Errors of Continuous Approximations of Adam Now we empirically verify the continuous approximation of Adam obtained from this framework. Firstly, we give the discretization error curves of the continuous approximations of Adam with discretization error to $\mathcal{O}(\eta)$, $\mathcal{O}(\eta^2)$, and $\mathcal{O}(\eta^3)$ under different combinations of hyperparameters. We demonstrate that the convergence points of the continuous approximations are getting closer and closer to those of the original Adam optimizer as the order increases. Secondly, we utilize the results of the continuous approximation to show the effect of the hyperparameters on the location of the convergence point of Adam.

We verify the correctness of our results on a simple 2-d model $f(x; \boldsymbol{\theta}) = \theta^1\theta^2 x$, in which $\boldsymbol{\theta} \in \mathbb{R}^2$ is the model parameter, x is the input of training data. The loss function is then defined as $L = \frac{1}{2}(f(x; \boldsymbol{\theta}) - y)^2$, in which y is the output of training data. The set of global minima forms a manifold, i.e. lying on the hyperbola $\theta^1\theta^2 x = y$ for this task. We will demonstrate that the discretization error of our higher-order continuous approximation outperforms existing works.

The hyperparameters of Adam, α and β, both take values in the range of $[0, 1)$. In order to stabilize the training process, we choose large α and β (typically 0.9, 0.99, or 0.999) in practice, so that the current parameter $\boldsymbol{\theta}_k$ has very limited influence on $\mathbf{m}_{k+1}$ and $\mathbf{v}_{k+1}$ when k is large, which is conducive

to the optimizer quickly getting rid of the local extremes and saddle points. This configuration of hyperparameters is also consistent with the practical usage of Adam in deep learning. And $\varepsilon = 10^{-3}$ is used in the experiment to avoid numerical errors caused by arithmetic operations.

In our experiments we compared the following trajectories and discretization errors: the discrete Adam optimizer; the continuous signGF, i.e., the first-order continuous approximation of Adam (Ma et al., 2022); the second-order continuous approximation of Adam first obtained by Cattaneo et al. (2024), with which our conclusion agrees; and lastly, the third-order continuous approximation of Adam obtained by our framework. One of the experimental results is shown in Fig. 2. The

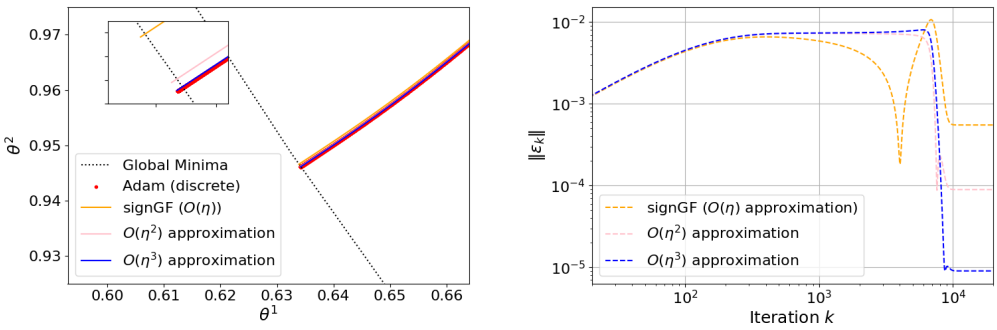


Figure 2: Discrete/continuous trajectories of parameters and discrete errors of Adam($\alpha = 0.9, \beta = 0.99, \varepsilon = 10^{-3}, \eta = 5 \times 10^{-5}$)

experimental results show that comparing with the $\mathcal{O}(\eta)$ and $\mathcal{O}(\eta^2)$ continuous approximations, the discretization error of our $\mathcal{O}(\eta^3)$ continuous approximation is reduced for all learning rates, both in the case of $\alpha > \beta$ and $\alpha < \beta$. This suggests that our framework is indeed effective in providing higher-order continuous approximations of Adam.

More Experimental Results For the discretization error of our derived third-order approximation of Adam, we also conducted several sets of numerical experiments for different parameter combinations. The relevant experimental results are as follows.

The first set of experiments is for the usual $\alpha > \beta$ case, taking $\alpha = 0.99$, $\beta = 0.9$, and the other experimental parameters are the same as experiment in Section 3.1. The discretization error of the continuous approximations of Adam under different learning rates η is shown in Fig. 3.

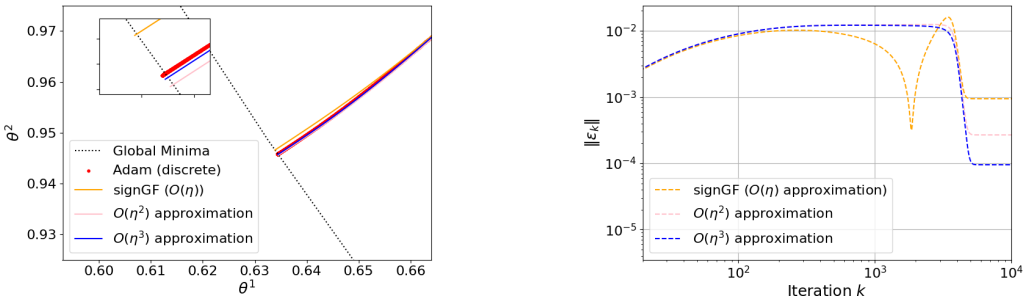
The second set of experiments is for cases that $\alpha < \beta$. We take $\alpha = 0.9$ and $\beta = 0.99$, the discretization error of the continuous approximations of Adam under different learning rates η is shown in Fig. 4.

It can be seen that the discretization error of the third-order continuous approximation of Adam obtained by us is better than that of the signGF and the second-order continuous approximation obtained by Cattaneo et al. (2024) for various hyperparameter combinations, which further improves the reliability of our result for the third-order continuous approximation of Adam.

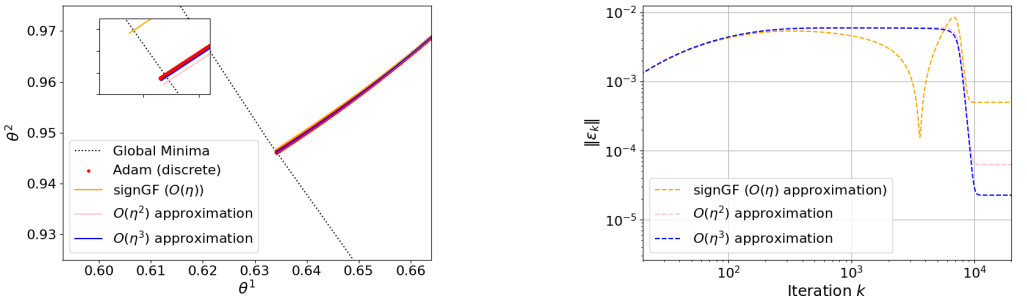
C.2 Details for Section 3.2

For the control group HB optimizer, we set $\eta = 0.1$ and $\beta = 0.9$. For all Adam optimizers in Experiment 1, we set $\eta = 0.001, \alpha = 0.999, \beta = 0.9$ in common, only ε is variable. And in Experiment 2, we set $\eta = 0.01, \alpha = 0.999, \beta = 0.9$ for all Adam optimizers.

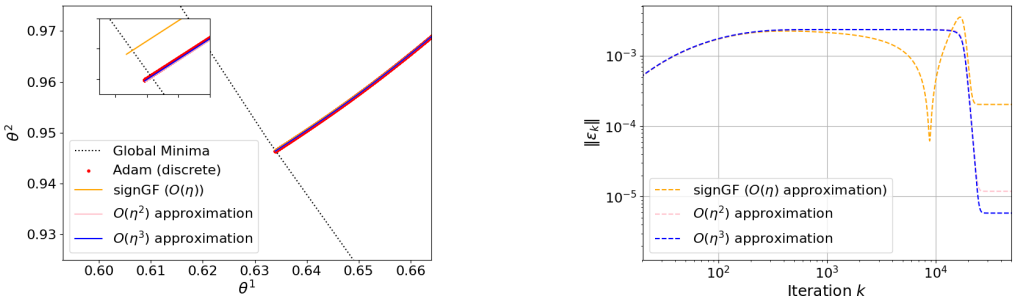
Experiment 1: Regularization of GD, HB and Adam In order to verify the newly discovered regularization effects and regime transition phenomenon, we trained LeNet (LeCun et al., 1998) on the MNIST dataset using GD, HB and Adam. Due to the huge difference in the convergence rate of the three optimizers, we selected different hyperparameters to allow their decrease in loss at the same level. Fig. 5 shows our numerical experiment results, in which Fig. 5a shows that in the beginning of the training, our hyperparameter settings cause the loss of all optimizers to decrease at a similar rate,



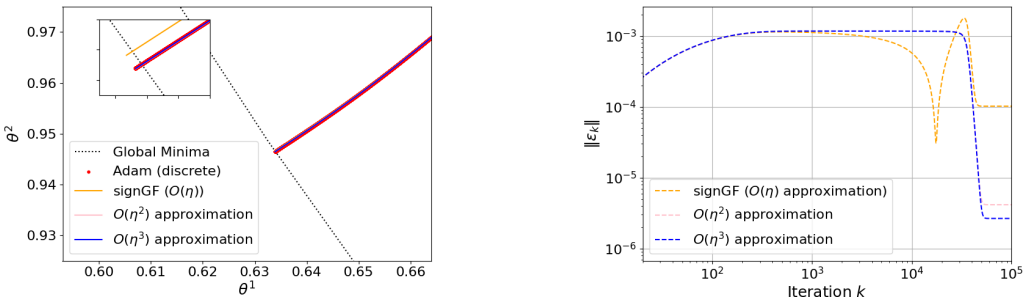
(a) Case 1: $\eta = 10^{-4}$



(b) Case 2: $\eta = 5 \times 10^{-5}$

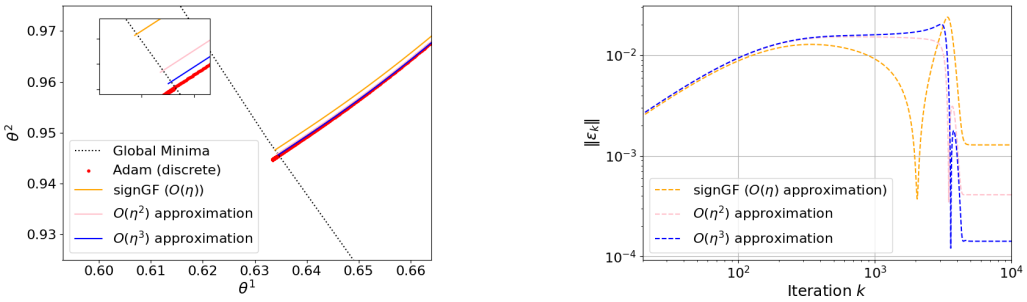


(c) Case 3: $\eta = 2 \times 10^{-5}$

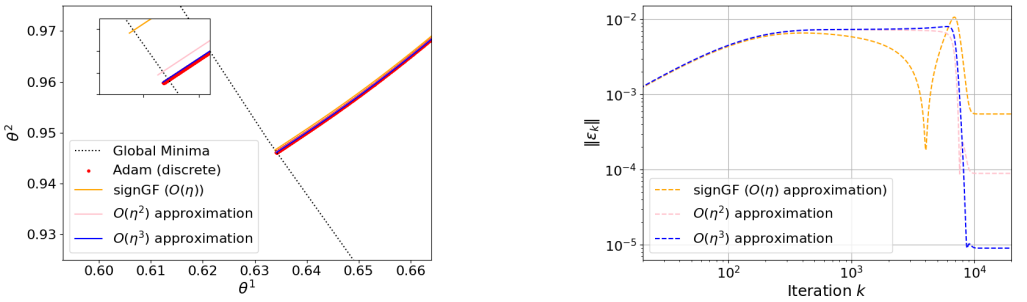


(d) Case 4: $\eta = 10^{-5}$

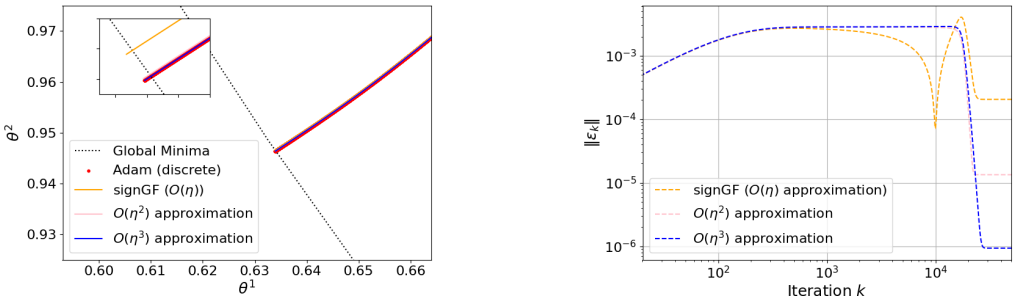
Figure 3: Discrete/continuous trajectories of parameters and discrete errors of Adam ($\alpha = 0.99, \beta = 0.9, \varepsilon = 10^{-3}$).



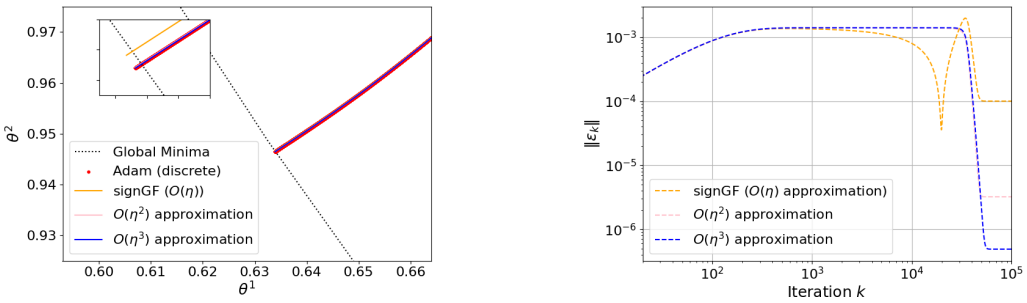
(a) Case 1: $\eta = 10^{-4}$



(b) Case 2: $\eta = 5 \times 10^{-5}$



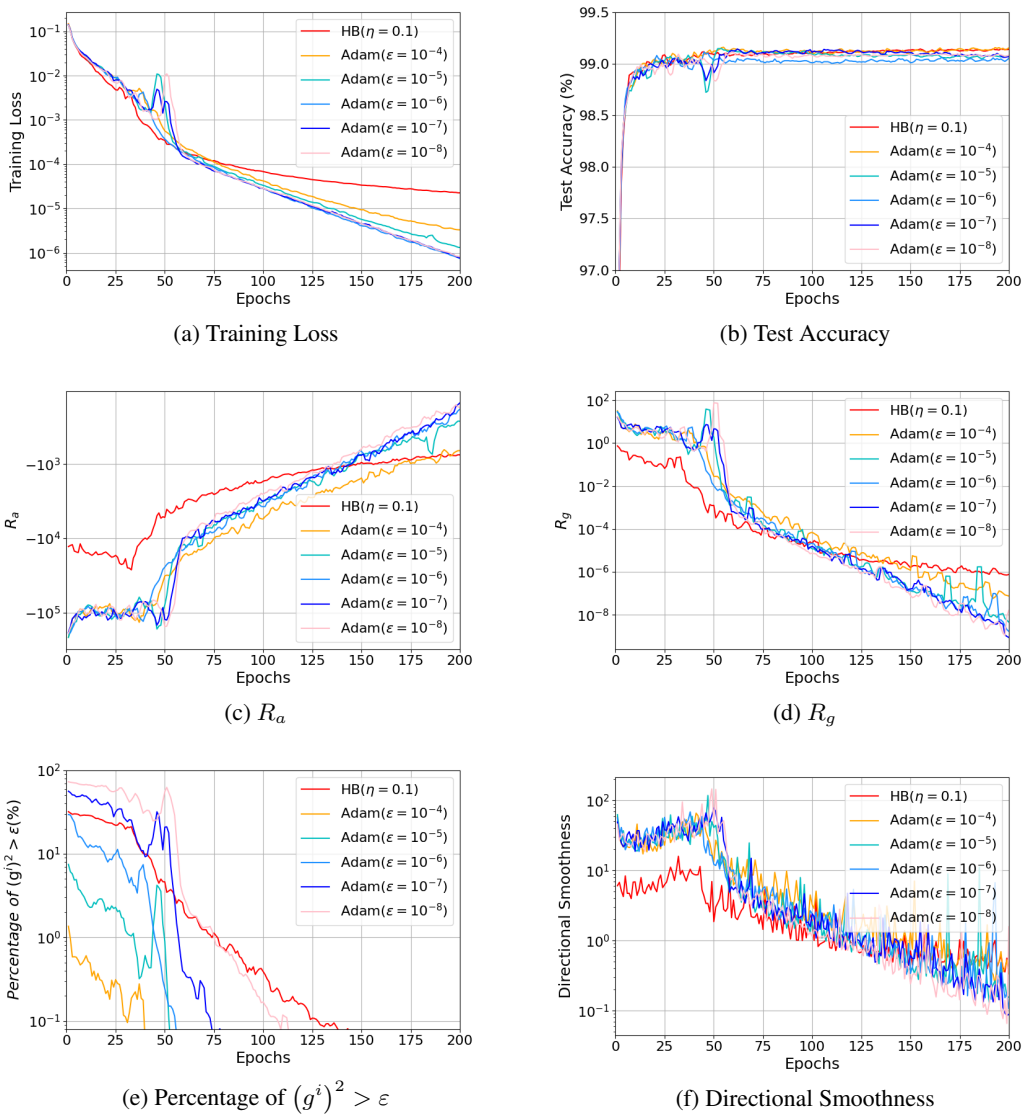
(c) Case 3: $\eta = 2 \times 10^{-5}$



(d) Case 4: $\eta = 10^{-5}$

Figure 4: Discrete/continuous trajectories of parameters and discrete errors of Adam ($\alpha = 0.9, \beta = 0.99, \varepsilon = 10^{-3}$).

which allows us observe the comparison of the behavior of different optimizers fairly. Numerical experiments confirm the regime transition of the third-order regularization for Adam, which behaves significantly differently from the regularization of GD and HB. In addition, our predicted regime transitions are also observed in the experiments.



(a) Training Loss

(b) Test Accuracy

(c) R_a

(d) R_g

(e) Percentage of $\left(g^i\right)^2 > \varepsilon$

(f) Directional Smoothness

Figure 5: Regularization in training process of different optimizers. ($\beta = 0.9, \eta = 0.1$ is taken in HB, and $\alpha = 0.999, \beta = 0.9, \eta = 0.001$ is taken in Adam)

Concretely, we first focus on Adam's behavior, and set an HB optimizer that uses a larger step size $\eta = 0.1$, resulting in a convergence rate comparable to Adam in this example, as a control group. In our experiment, each optimizer causes the model accuracy on the test set to reach near a stable value within nearly 20 epochs in Fig. 5b. For our main conclusion about the implicit bias of Adam, it can be seen from Fig. 5c that, significantly different from HB, in the first 50 epochs, Adam has a clear tendency to minimize R_a. On the contrary, Adam is insensitive to the regularization term R_g of HB in the early stages of optimization. When HB minimize R_g, R_g of the Adam optimizers do not change significantly in the first 50 epochs. However, at around 50 epochs, Fig. 5c shows that R_a of Adam optimizers increase in a short period and are no longer restricted, which is a noteworthy phenomenon. Adam henceforth no longer regularizes R_a as it did in the previous epochs, but instead exhibits a regularization similar to that of GD and HB in Fig. 5d. This is just in line with our prediction that

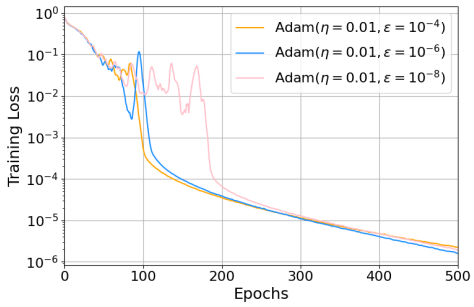
there is a regularization transition in Adam as the relationship between the components of the gradient g^i and ε changes, resulting in a gradual regime transition in the optimization process.

In addition to the change in regularization, our conclusion can be corroborated by two other facts. First, according to our prediction, if ε increases, the gradient will fail to satisfy $\left(g^i\right)^2 \gg \varepsilon$ earlier. Fig. 5c and Fig. 5d show the behavior of Adam optimizers with different ε, while not immune to the effects of randomness, in most cases the regime transition does seem to appear earlier as ε increases, which is consistent with our inference. Second, Fig. 5e illustrates how the relationship between $\left(g^i\right)^2$ and ε changes during training (in which the ε of HB was set to 10^{-8} and served only as a control group with no practical significance). It can be seen that the moment when the rapid decrease in the percentage of $\left(g^i\right)^2 > \varepsilon$ occurs coincides almost exactly with the moment when the regime transition of the Adam regularization occurs.

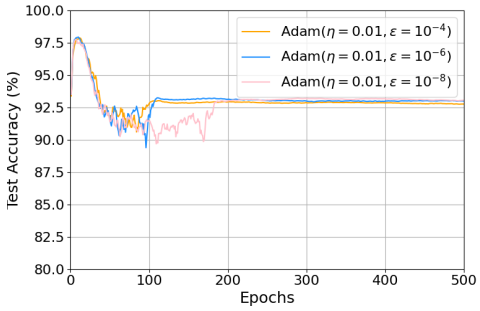
In order to show the difference between HB and Adam in terms of third-order regularization more clearly, we also draw the third-order regularization metric of HB, i.e., Directional Smoothness in Fig. 5f at the same time. We can see that the Directional Smoothness of HB always maintains a small value under the control of regularization, which confirms the correctness of the third-order regularization of HB. However, before the regime transition occurs, the Directional Smoothness of Adam is significantly larger than that of HB, which indicates that Adam has not implicitly regularized the Directional Smoothness. And after the regime transition occurs, Adam's Directional Smoothness quickly decreases to the same level as that of HB, which reveals that the implicit regularization of Adam is becoming the same as that of HB after the regime transition.

Experiment 2: Regime Transition of Adam under Different Parameters Due to the simplicity of the task on the MNIST dataset, we could not draw further conclusions about the effect of the regularization of Adam on its generalization ability directly from this experiment. Inspired by the study of Xiong et al. (2024) on SGD with random labeling noise, we add 10% random labeling noise to MNIST based on the experiment in this section to observe whether the behavior of Adam on this dataset is still consistent with our prediction and focus on the generalization performance of Adam. The corresponding experimental results are shown in Fig. 6. It should be noted that, since the fitting difficulty of the dataset with added noise is much larger than that of the original MNIST, in order to speed up the convergence, we use a larger learning rate of $\eta = 0.01$ in this experiment.

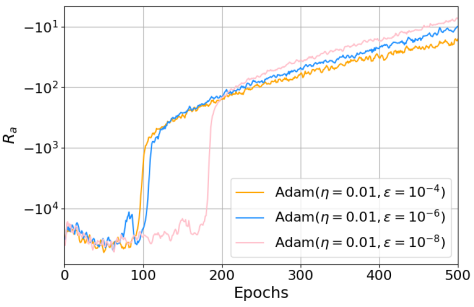
The results of Fig. 6 show that the regime transition of Adam still exists in this experiment, which further indicates that our conclusion is reliable. Without repeating the same conclusions as in the previous paragraphs, we will focus on the generalization ability of Adam in relation to the regime transition. As can be seen from Fig. 6, the regime transition of Adam corresponding to the hyperparameters $\varepsilon = 10^{-4}$, $\varepsilon = 10^{-6}$, and $\varepsilon = 10^{-8}$ occur around the 90th, 100th, and 180th epoch respectively. Notice the change of test accuracy shown in Fig. 6b. It can be seen that the test accuracies of Adam optimizers with three different hyperparameters increase rapidly to their maximum throughout the training process within the first few epochs. Subsequently, as the optimizers start to overfit on the random noise labels we added, the test accuracies of the models drop to around 91%, although the training losses are still decreasing (Fig. 6a), which is in line with the common machine learning theory. However, when the regime transitions of the optimizers occur, the test accuracies of these models rebound even though the training loss is still decreasing, which implies that the models have regained some generalization ability. This phenomenon corresponds to our judgment that the regularization behavior of Adam changes as it approaches the minima. This experiment further justifies our understanding of the different roles of regularization before and after the regime transition of Adam, i.e., the regularization R_a drives the optimizer to accelerate convergence before the regime transition, while the regularization R_g drives the optimizer to generalize better after the transition.



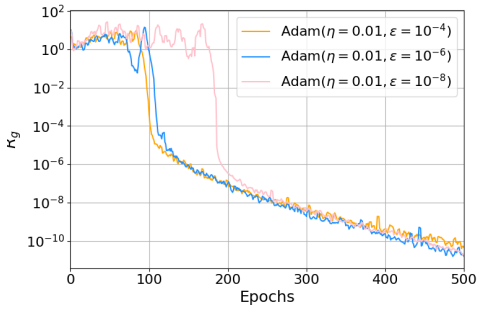
(a) Training Loss



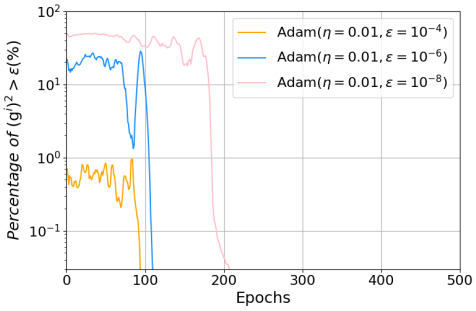
(b) Test Accuracy



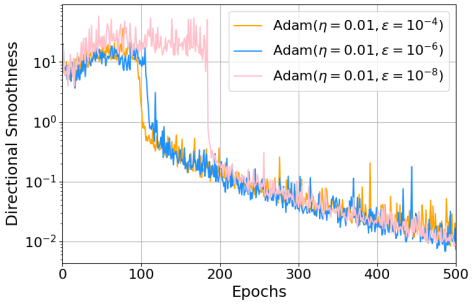
(c) R_a



(d) R_g



(e) Percentage of $\left(g^i\right)^2 > \varepsilon$



(f) Directional Smoothness

Figure 6: Regularization in training process of different optimizers. ($\beta = 0.9, \eta = 0.1$ is taken in HB, and $\alpha = 0.999, \beta = 0.9, \eta = 0.001$ is taken in Adam)